LAPG 7122.1
Effective Date: November 9, 1999
Expiration Date: November 9, 2004

Langley Research Center

SYSTEMS ENGINEERING HANDBOOK FOR IN-HOUSE SPACE FLIGHT PROJECTS

(Due to formatting problems, this edition of LAPG 7122.1 contains errors in page numbering, figure location, and general format style. However, the content of the document is correct and represents the Center's policy for and implementation of the Systems Engineering Process.)

National Aeronautics and Space Administration

PREFACE

This handbook provides a summary of the systems engineering procedures associated with space flight projects where the primary project components (hardware and software) are developed in-house. The information in this handbook should serve as a basic reference for projects to develop a tailored sequence of events which will lead to achieving the best system design for the project.

This handbook is a supplement to the Langley Management Manual and is primarily applicable to space flight projects which are implemented in-house at Langley Research Center (LaRC). However, the fundamental systems engineering disciplines described will be applicable to early studies for contracted projects and to aeronautical projects and ground facility developments.

Revisions and additions to this handbook will be issued annually when changes or refinements in the systems engineering process or its implementation at LaRC are required.

Glenn R. Taylor
Acting Chief, Aerospace Mechanical Systems Division

DISTRIBUTION:
SDL 065 - Group Directors, Offices Reporting to Group Directors, and Division Chiefs
157/Information Systems Division (20 copies)
288/Space Systems and Concepts Division (10 copies)
356/Space Projects Office (10 copies)
401/Atmospheric Sciences Division (10 copies)
421/Office of Safety, Environment and Mission Assurance (10 copies)
430/Aerospace Mechanical Systems Division (120 copies)
476/Aerospace Electronic Systems Division (60 copies)
123/Institutional Support Branch, LMD (10 copies)

TABLE OF CONTENTS

TABLE OF CONTENTS - Continued

TABLE OF CONTENTS - Concluded

INTRODUCTION

SUMMARY

This Chapter discusses the scope of small space flight projects at Langley Research Center and describes how the systems engineering process is integrated with the Langley project life cycle to achieve project goals in a systematic way.

1.1 PURPOSE

The NPG 7120.5A, NASA Program and Project Management Processes and Requirements, states that a clearly structured and defined systems engineering process will be used in the formulation and implementation of NASA projects to assure that each system achieves its scientific and/or technical goals with demonstrated quality performance and within planned budget and schedules. This handbook describes the systems engineering discipline and tools available to support the advocacy, definition, and development of flight systems at Langley Research Center.

However, the information in this handbook should serve a basic reference for any project to develop a tailored sequence of events, which will lead to achieving the best system design for the project.

This handbook provides a reference for project team members in selecting, tailoring, and implementing a systems engineering process for LaRC projects. This handbook is intended to provide sufficient treatment to allow a comprehensive application of systems engineering for the largest and most complex projects. Depending on the scope of each effort, the process should be tailored to assure the appropriate level of systems engineering application. These determinations should be made at project inception through discussions between the systems engineer and the project or study manager, and, upon approval by Center management, be made a part of the Project Plan. Project personnel may find the handbook useful in defining and planning detailed tasks within the systems engineering context. The handbook also provides a review of related computer tools and glossary terms.

1.2 SCOPE

The procedures described are primarily applicable to small space flight projects that are implemented in-house at LaRC, as defined in LAPD 7120.2, Authority and Responsibilities of Managers of Small Space Flight Projects. In such systems, the primary project components (hardware and software) are developed and integrated in-house, although elements may be contracted to industry. It is intended that this handbook provide the steps necessary for the application of effective systems engineering to these developments.

Similar procedures are used when a system is acquired out-of-house, but this handbook does not address contractor-specific issues. However, even contracted efforts will usually require an in-house systems study to evaluate the progress and results of the contract effort. As a general rule, formulation sub-process studies will be conducted in-house, regardless of the overall acquisition strategy.

Often, a LaRC small space flight project may be part of a larger NASA project managed at another Center. If the other Center requests it, the LaRC project may have to conform to the larger system's requirements.

NASA policies and responsibilities specified in NPG 7120.5A, delegate responsibility for the implementation of a project, including systems engineering and design, to the Project Manager within guidelines and controls described by the funding organization and the LaRC Center Director. This handbook addresses all aspects of systems development, including the creation of hardware and software architectures, and the development and management interfaces between subsystems, from the initiation of the project through flight operations and termination.

New projects are initialized from promising advanced studies or in response to a NASA headquarters request or an Announcement of Opportunity. During the formulation sub-process a study team is formed, at the request of the sponsoring organization, to assess the feasibility of the proposed effort. The systems engineer and the study manager implement the systems engineering process to achieve the goals and present the recommendations of the study team to the sponsoring management.

If approval is given to continue with the sub-process with iterative formulation, a project manager will typically be assigned to direct the expanded effort by the study team. It will be the responsibility of the study team to finalize the recommended Project Plan including a systems engineering management plan defining the systems engineering process to be used. It is intended that the project manager and the systems engineer consider all the systems engineering activities presented in this handbook; however, it is their responsibility to tailor these recommendations as appropriate to their project.

1.3 BACKGROUND

Efforts to implement an effective systems approach to NASA space flight programs are in progress Agencywide. These efforts have resulted in NPG 7120.5A, NASA Program and Project Management Processes and Requirements, and SP-6105, NASA Systems Engineering Handbook. This handbook is consistent with NPG 7120.5A but was tailored for the systems engineering process at LaRC. A selected list of NASA publications is contained in Appendix E.

A system may be defined as a set of interrelated elements organized to work together toward a common goal. Systems may be considered to be the building blocks which comprise projects and programs. Systems can be decomposed into smaller entities that may be considered systems within their defined boundaries. It is this subsystem decomposition, along with careful and complete boundary or interface definition, which allows the partitioning of complex systems into manageable entities. Ultimately, these manageable pieces are represented by those drawings and specifications necessary to construct the operational system, and thus satisfy system technical goals. In addition to these technical aspects, project goals include non-technical goals, such as cost, schedule, and Agency interests. It is the responsibility of the project to develop a system that satisfies the principal customer, while also satisfying ancillary project constraints. Many constraints are external, and out of the control of the project. However, systems engineering has a strong impact on the cost and schedule projections which are the basis for funding, and is accordingly responsible for their accuracy.

A hierarchical system reference is often used for decomposing hardware and software systems down to the lowest level. Such a hierarchy can be used as the basis for the Work Breakdown Structure (WBS) and for subsequent systems decomposition and allocation of requirements. The systems hierarchy shown in Table 1.1 has been proposed as a common NASA terminology for hardware and software system constituents. Each system must be structured with the number and type of levels appropriate for the specific application and consistent with the system external interfaces.

1.4 SYSTEMS ENGINEERING PROCESS

The objective of systems engineering is to provide a robust system that satisfies the customer's technical performance objectives within the constraints of cost and schedule. The challenge is to ensure that a system is developed which meets all imposed requirements and provides the proper balance of system performance, life cycle cost, and development time.

A systems engineering process is the approach to achieve this objective. This process is an orderly sequence of tasks designed to accomplish the optimum design of systems. Optimum does not imply the ultimate in technical performance, the lowest cost, or the shortest schedule. Rather, optimum is defined by the project goals and requirements and will be a balance of these three factors. The systems engineering approach includes the identification of scientific and technical goals; the definition of system constraints and performance measures; the analysis and development of system concepts; and review, verification, and validation of the candidate system(s). During the formulation phases of the project, the systems engineering process relies heavily on an iterative, analytical approach termed the Systems Analysis and Design Procedure. The steps of this procedure are outlined in Table 1.2 and will be described in more detail in Chapter 3. The procedure for applying structured systems engineering to LaRC in-house projects is discussed in Chapter 4 and outlined in Chapter 6.

Table 1.1 - System Hardware and Software Hierarchies

SYSTEMS HIERARCHY

Program
Project
System
Segment

Software	Hardware
Computer Software Configuration Item (CSCI)	Element
	Subsystem
Computer Software Component (CSC)	Assembly
Computer Software Unit (CSU)	Subassembly
	Part

Table 1.2 - Steps in the Systems Analysis and Design Procedure.
1. Initialization: The initial step is to organize the study and acquire the required tools and resources support. In a LaRC development, this step assures that new projects have been reviewed by the Sponsoring Group Director and that appropriate resources and skills are available to achieve the goals of the current project phase. Typically, the start of a new project phase requiring NASA Headquarters funding will be closely related to the NASA budget cycle. Important systems engineering factors to be addressed

	include an assessment of the objectives for the phase, technical approaches, and analytical tools to be used.
2.	User Needs and Goals Analysis: The major product of this step will be the Goals Analysis Document and Hierarchy, which will initially be in a preliminary form and will gradually be refined into subsequent phases. The goals document is the basis for subsequent system metrics such as performance measures, requirements flow down, and verification and validation standards. The systems engineering function must monitor system performance through all project phases to assure that system goals will be achieved.
3.	Systems Requirements and Constraints: This step establishes the constraints at each phase leading to progressively more detailed definition of requirements for the hardware and software architectures. During the early phases, system requirements are developed which satisfy the system goals. As the system design progresses, hardware and software requirements are allocated to the segments of the system and progressively lower in the hierarchy. An important systems engineering function is to track the allocated requirements through final verification and maintain traceability to the system requirements.
4.	Performance Measures: Performance measures are parameters that are defined to provide criteria for subsequent systems analyses and trade studies. For product developments, these are the variables used to judge the overall attractiveness of system candidates and a subset of the project requirements. Initially, performance measures are the parameters used for system selection and then they evolve, along with constraints, into requirements or systems validation criteria.
5.	Systems Concepts: Alternative solutions are generated in this step. Initially, the output of this activity is a set of candidate system options for analysis and trade-off studies. In Phase B, the list of alternate concepts is reduced and the baseline system concept is selected. Requirements and resources are allocated and "design to" specifications are defined for the segments. In project implementation, these segment requirements become the standard for the application of systems engineering analysis at the segment level. This decomposition process is repeated throughout the systems hierarchy until concepts are defined down to a level which allows for easy piece part selection and computer software unit development.
6.	Concepts Analysis: This step analyzes and defines the performance of alternate systems concepts so that comparisons can be made for final system concept selection. Mathematical modeling and computer simulation are frequently employed during this step. Analytical tools are used throughout the project life cycle to estimate and verify systems performance.
7.	Concepts Ranking: During this step the set of systems concepts are ranked by decision analysis in order of their overall performance (including cost and

schedule) so that selection of preferred concepts can be made. The resulting Alternate Concepts Analysis Document becomes the basis for the configuration management plan that is utilized throughout the project life cycle.
8. Systems Development: This step incorporates all of the detailed development activities required to advance the systems concept(s) and design to the desired level of maturity for the current phase of the project. These activities include design refinements and risk reduction activities that were identified during concept analysis.
9. Review, Verification, and Validation: The Systems Analysis and Design Procedure is iterative and requires ongoing, critical examination of efforts. During this step, the systems concepts are subjected to verification processes to assure system integrity and to external reviews to secure management concurrence and commitment for the project to continue. The verification process is employed throughout the project life cycle.
10. Decision Point: This important step or control gate will provide management direction for the future of the project; approval to proceed to the next phase, direction to go back and revise the project approach, or a decision to terminate the project effort.

1.5 PROJECT LIFE CYCLE

A new project is born out of recognition of a need or opportunity which addresses NASA's goals and missions. If the project appears to be promising, it is selected by the LaRC sponsoring group for further definition and a Pre-Phase A system study is undertaken to evaluate feasibility. A typical project passes through defined phases or cycles as it proceeds from conceptual definition through design, fabrication, integration, and test to an operational role. Progression from one phase to the next is dependent upon meeting the phase objectives and passing prescribed reviews and decision points. The total sequence is termed the "project life cycle." The progressive phases in the NASA project life cycle are listed in Table 1.3. The life cycle may be tailored to meet the unique requirements of a specific project.

Table 1.3 - NASA Project Life Cycle

1. Pre-Phase A: ADVANCED STUDIES - Preliminary Requirements and Concepts Analysis
2. Phase A: PRELIMINARY ANALYSIS - Requirements Definition and Conceptual Trade Studies
3. Phase B: DEFINITION - Concept Definition and Preliminary Design (Source selection, if required)
4. Phase C: DESIGN - Final Design and Engineering Development
5. Phase D: DEVELOPMENT - Fabrication, Integration, Test, and Evaluation
6. Phase E: OPERATIONS - Preflight and Flight Mission Operations and Disposal

1.6 SYSTEMS ANALYSIS AND DESIGN PROCEDURE MODEL

The Systems Analysis and Design Procedure addresses the detailed activities and products which must be accomplished to support project decisions concerning the system under development. Systems engineering analysis is inherently an iterative process resulting in successive refinement of the system design through each phase of the project. The Systems Analysis and Design Procedure is heavily utilized by the project team in the early formulation phases (Pre-Phase A, Phase A, and Phase B) of the project, and is used in support of subsystem level trades as the development progresses. In the implementation phases (Phase C, Phase D, and Phase E) of the project, systems engineering is more heavily involved with integration and verification activities and the Systems Analysis and Design Procedure is less formalized. In the closing activities of the implementation phases, systems engineering attempts to validate system performance against customer requirements and record lessons learned from the project. It is important to note the clear distinction between systems engineering, which is concerned with all aspects of system development, and the Systems Analysis and Design Procedure, which is a tool utilized by engineers for optimization.

The steps of the Systems Analysis and Design Procedure are combined with the project life cycle in the schematic model shown in Figure 1.1. The process starts at the center of the circle when Pre-Phase A activities are initiated and follows a clockwise path through each sector of the circle as each task of the Systems Analysis and Design Procedure is addressed.

When the circuit is completed the option exists to repeat the cycle (or specific tasks) for better definition or to pass through the decision point to the next phase of the project. Each formulative phase of the project is addressed using the same systematic approach. As the project moves from phase to phase, the tasks also evolve to address the changing objectives as will be shown in Chapter 6.

This Systems Analysis and Design Procedure model has several distinctive features that can help in visualizing the overall project development process:

- The process is initiated by a formal management decision to select a promising candidate concept or idea for further study and investigation. There may be many research proposals under consideration at any given time, but only a few can be developed into potential projects. When a candidate project is advocated by management for in-depth study, this establishes priority, level of effort of personnel, and associated schedule for a systems study.

- The model illustrates a specific sequence of steps to be followed during the systems study. The process steps are completed consecutively in a logical progression during each phase of the study, as appropriate. This provides a focused and structured method that will result in the most efficient approach to the study. It is expected that the Systems Analysis and Design Procedure will be guided by the systems engineer, who will direct and coordinate the effort to assure that the study goals are achieved.

- The process steps outline a problem-solving approach rather than a fixed step-by-step procedure. Each step may be tailored to accomplish the defined objectives of the current phase of the project. For example, during the early formulative phases of the project (Pre-Phase A, Phase A, and Phase B), the emphasis is on system level analysis and design tasks. In the later implementation phases (Phase C and Phase D), the focus is on detailed design of segments and lower levels of assembly. The steps may be repeated iteratively to achieve the design maturity desired during the phase. In the event of a change in project goals, the process may be repeated to define the impact of changes.

- Each phase of the process is a distinct entity as indicated by the concentric, labeled circles. There may be circumstances in a project where certain areas lead others; for example, long lead items which require longer to develop. In general, the project should remain focused on the objectives of each particular phase until that phase is completed. As noted on the model, transition from one phase to another requires passage through a decision point or control gate. This assures management concurrence and support. Each phase is a distinct activity and successful completion of a phase demonstrates that the project is showing progress toward accomplishing its end goals.

The general flow of the systems engineering process for a hypothetical LaRC project is displayed in Figure 1.2. This figure shows how the systems engineering process and the Systems Analysis and Design Procedure may be applied within the context of project life cycle phases and control gates. The generic steps of the procedure including goals analysis, systems requirements, concepts analysis and ranking, development, and so forth, are shown within each phase. The iterative nature of the process is also emphasized by the provision for repeating the cycle for improved resolution of the products. The process is repeated in subsequent phases as applicable with appropriate changes in emphasis to continue refinement of the system. This evolutionary approach will be developed in more detail in Chapter 4.

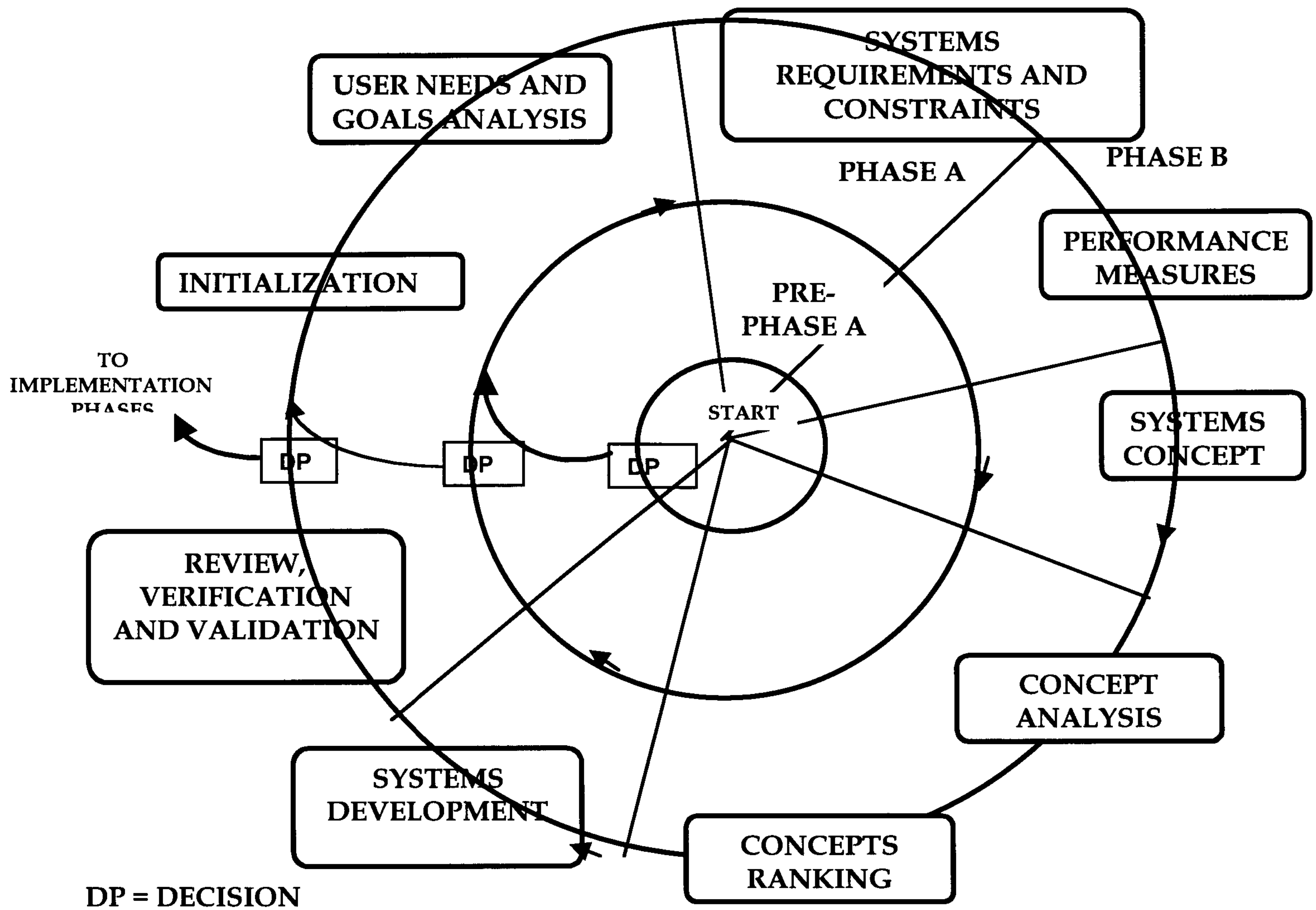

Figure 1-1 – Systems Analysis and Design Procedure Model.

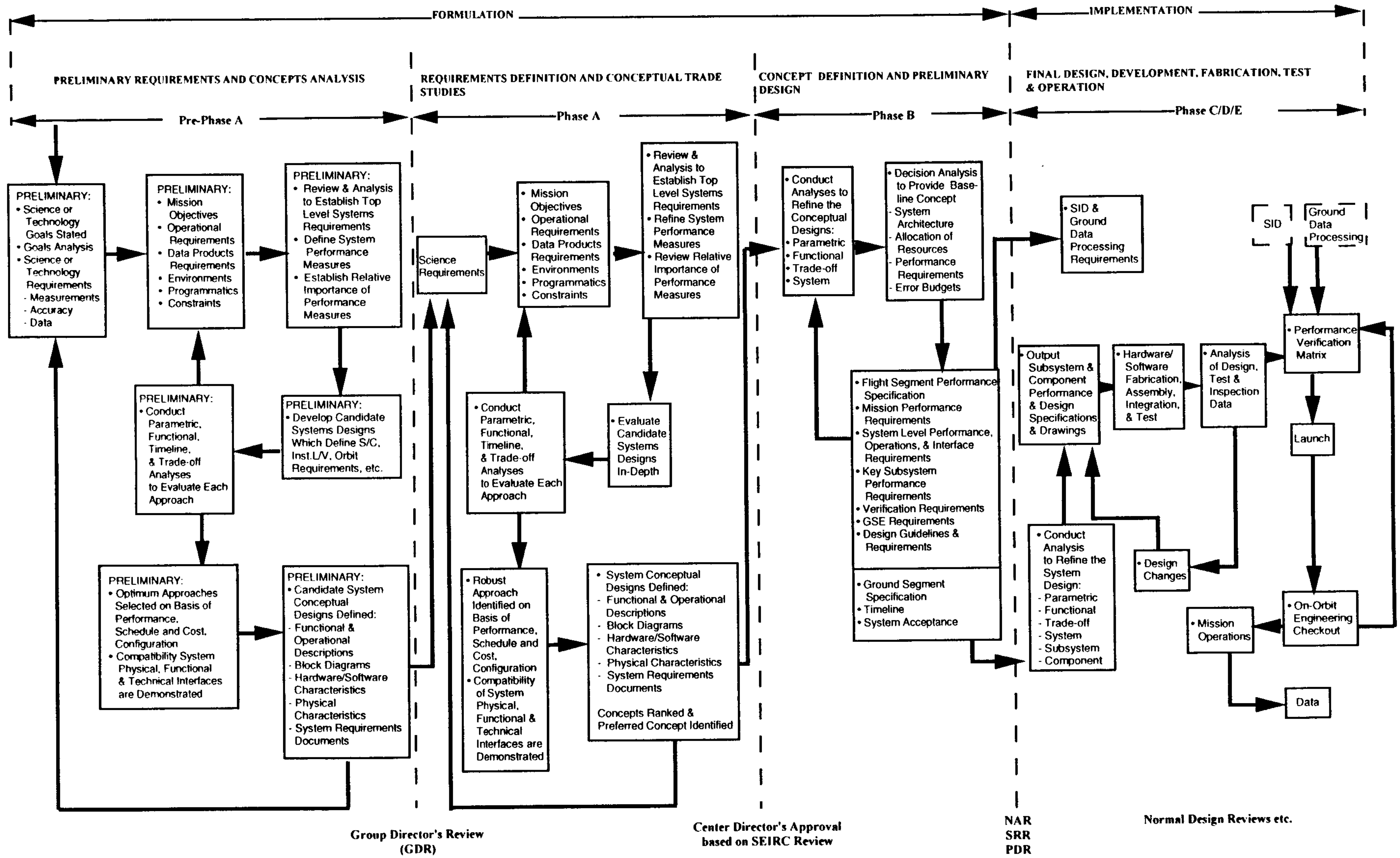

Figure 1.2 - Flow of the Systems Engineering Process

Chapter 2

SYSTEMS ENGINEERING MANAGEMENT

SUMMARY

This Chapter gives an overview of the interaction between project management and systems engineering in the Langley Research Center project environment.

2.1 PROJECT MANAGEMENT

Within the NASA context, the Project Manager is defined by NPD 7120.4A, "Program/Project Management," as "the senior official at the NASA field installation exclusively responsible for managing execution of the project life cycle to accomplish program objectives within guidelines and controls prescribed by program and field installation management." The Langley Research Center (LaRC) Project Manager reports through Center management to the Program Manager at NASA Headquarters or at the host field installation which has overall program responsibility. Thus, the Project Manager has the key responsibility to be the LaRC focal point for the project.

A major responsibility of project management is to control the project resources within the three constraints of time, cost, and performance. This task is ultimately the responsibility of the Project Manager. The Project Manager also is responsible for developing the overall Project Plan and for the day-to-day responsibilities of managing the project team to achieve the goals that have been defined.

One role of the Project Manager (or Pre-Phase A Study Manager) is to establish a project environment which is supportive of the systems engineering task. This role includes setting project priorities and ensuring needed resources are available. The Project Manager is also responsible for resolving problems identified by the systems engineer and project team. At the beginning of each phase of the project, the Project Manager, with the systems engineer, must prioritize the systems engineering tasks and define the project objectives for that phase.

2.2 SYSTEMS ENGINEERING RESPONSIBILITIES

Within the project organization, the Project Manager typically delegates systems engineering and technical responsibility to the systems engineer. In this position, the systems engineer has technical direction over all the engineering disciplines and must coordinate all project activities within the context of the systems engineering process. Just as the Project Manager's focus is primarily on overall project management and controlling the project resources, the systems engineer's emphasis and responsibility is to assure that the system accomplishes its technical purpose in the most cost-effective way. The Project Manager's role and the systems engineer's role are mutually dependent and the effectiveness of the project effort hinges on their interaction.

At LaRC, the systems engineer's role will be assigned to different individuals, depending upon the project organization. The systems engineer may have a title such as Instrument Manager (IM), Technical Project Engineer (TPE), Instrument Project Engineer (IPE), or Systems Engineer. The term "systems engineer," as used in this handbook, describes the individual responsible for managing the systems engineering process.

A study or project will usually have only one systems engineer assigned during all project phases. However, larger projects which require a broad range of technical expertise may need systems engineers assigned to different segments of the system. All of these individuals should be familiar with the systems engineering process as implemented on the project.

The systems engineer has overall technical accountability for the design and operation of all systems on the project. Detailed project activities are actually performed by the cognizant project discipline, that is, design engineering, test engineering, software engineering, fabrication, assembly, and so forth. In terms of the three project restraints of time, cost, and performance, systems engineering is primarily responsible for system performance. System performance also impacts time and cost; therefore, the systems engineer must continually address all three project restraints.

The systems engineer is responsible to the Project Manager and is usually the focal point of the project for all technical performance at the systems level. The Project Manager specifically addresses project management interfaces and specialty functions such as cost, schedule control, safety, program assurance, and configuration management. The systems engineer is concerned with all the aspects of systems engineering management including: baseline management, requirements review and traceability, system specifications, change control, design reviews, audits, document control, failure review boards, control gates, and performance certification. The systems engineer is also concerned with the internal and external system technical interfaces. It is the systems engineer's responsibility to assure that all of the systems perform properly when the system is fully integrated.

The systems engineer must have broad experience in many technical disciplines and should have relevant systems experience with flight hardware and software in the project environment. The systems engineer must also be able to work effectively with the project team to accomplish the systems engineering tasks for each project phase. The composition of the project team by phase will be discussed in detail in Chapter 4. The

the project team by phase will be discussed in detail in Chapter 4. The systems engineer may also serve as the Contracting Officer's Technical Representative (COTR) for contracted efforts.

2.3 SYSTEMS ENGINEERING MANAGEMENT

This handbook is to be used as a guide for each project to provide a planned application of systems engineering which is appropriate to the effort. The systems engineering management plan is the project document that defines how the project systems engineering function will be technically managed within the constraints established by the Project Plan. For a large project, a formal, documented systems engineering management plan may be used to describe how the systems engineering effort will be managed. The document may be organized in three parts:

Part I - Technical Program Planning and Control

The first part defines organizational responsibilities and addresses such issues as configuration, documentation, design control methods, and review requirements.

Part II - Systems Engineering Process

The second part describes the process to be used, risk management approach, types of mathematical simulation models, and other application related information.

Part III - Engineering Specialty Integration

The third part describes the integration of the specialty engineering disciplines, project approach to concurrent engineering, verification and validation, and so forth.

Chapter 3

SYSTEMS ANALYSIS AND DESIGN PROCEDURE

SUMMARY

This Chapter describes the overall philosophy of the Systems Analysis and Design Procedure with details on application during the critical, early Formulation Phases and in the later Implementation Phases of the project.

3.1 INTRODUCTION

This Chapter describes the overall philosophy of the Systems Analysis and Design Procedure; details on implementation in each phase will be provided in Chapter 6. The process is a simple set of steps, applicable to any problem, and is designed to find the best solution, versus an adequate solution. "Best," in this context, is from the point of view of technical performance, as well as cost and schedule. The process begins deductively, from the general to the specific, by establishing the broad top goal of the project and working toward the details. As the project progresses, "bottom-up" work is accomplished to compliment the initial deductive nature. However, the fundamental concept requires that the overall picture is established first, before detailed actions (for example, hardware piece part or software module design) are undertaken. The goals established initially in the process serve as a clear direction for project work and establish the criteria for success in the form of constraints and variables for optimization, typically referred to as requirements. After establishing these performance requirements, options are explored and ranked numerically. All facets of the process are repeated, or iterated, for error control, leading to successive refinement of systems goals and requirements. The following sections describe the Systems Analysis and Design Procedure in detail.

3.2 INITIALIZATION

This precursor step is necessary to assure that appropriate resources and skills are available to achieve the immediate phase goals within the specified schedule and cost constraints. It is necessary that the systems engineering function be properly planned prior to initiation of the study. Relevant factors include an assessment of the goals for the study, organizational responsibilities, technical approach, and analytical tools to be used. It is the responsibility of the systems engineer to assure that the study goals will be achieved on a timely basis. Any expected deviations from these goals should be immediately brought to the attention of the sponsoring Group Director.

Products of the initialization step are typically work and staffing plans and detailed schedules to meet the near term milestones. Formal decision points (control gates) may be supplemented by more detailed entrance criteria to be satisfied before work can commence at the start of the cycle. Exit criteria, in the form of required activities and products, may also be established to assure that key elements will be completed prior to the end of the Systems Analysis and Design Procedure cycle for the phase.

3.3 USER NEEDS AND GOALS ANALYSIS

This endeavor is perhaps the most crucial undertaking of the project, since it is here that success is defined. However, it is insufficient to simply define success, because the definition is unclear or ambiguous. The steps of the goals analysis are designed to provide a coherent, complete set of objectives for the project team. These steps are:

3.3.1 Top Level Goal

The first concern of the goals analysis is to obtain a single, top level goal for the undertaking. This should be a short, concise statement of what the project hopes to accomplish and should exclude, to the greatest extent possible, any details as to how it will be achieved. "How" qualifiers are, in effect, constraints that limit the possible alternatives. Typically, the top level goal is available at the onset; but it is still necessary to assure the accuracy and validity of the project goal statement. This is best accomplished by putting this goal in context with other broader goals of the organization such as vision statements, policy, or organizational thrusts. Care should be taken to assure that this goal is consistent with other endeavors and to clearly emphasize that the project is addressing a new area of concern. Discussions should be conducted with the customer/user of the output to define the central issue of the project and to assure that project goals are consistent with defined mission needs. Typical considerations may be:

*Why is the goal as stated?
*Are there underlying goals not brought to light?
*Is the statement the true goal or rather a means of achieving another goal?

The purpose of this activity is to assure that the true goal is defined. The total project team should be a part of these discussions to assure a thorough understanding, but ultimately, the definition of the goal statement is the responsibility of the customer/user or, typically, the Principal Investigator. The systems engineer typically serves as a facilitator or coordinator in establishing the top level goal. Participation of a systems engineering facilitator acting as a third party in these discussions is beneficial since queries may be made in a non-threatening manner by a third party with no stake in the outcome.

3.3.2 Current Status

Once the top level goal statement is established, research is conducted to determine and record the current state of the art or status of efforts addressing the goal. Other current work which relates to the project, both internal and external to the organization, is recorded. A history of the events leading to the current situation is also helpful in preventing a repeat of past mistakes. Specific attention should be paid to the problems or shortcomings currently associated with the endeavor, along with the efforts under way to resolve those problems. Usually, the customer is the best source of this information, which may easily be obtained through discussions. However, further independent attempts to gather data are well advised as they may reveal information from unexpected sources. This may be done through library searches or by contacting identified external sources directly to query their knowledge of the situation. Conversations with those associated with similar efforts provide an excellent means for establishing a complete picture of the current state of the art. Emphasis should be placed on positive and negative aspects of the status quo. Any and all pertinent details are compiled in order to prevent duplication of effort and give the team a good understanding of the starting point of the project. The systems engineer should summarize this information in the Statement of Project Status. Given a firm foundation of the current standing, the next step follows directly.

3.3.3 Vision

The project vision is a set of statements that describes the desired situation upon successful completion of the project. This narrative describes the positive aspects of the current status, plus improvements made and a report of those problems or shortcomings which have been overcome. These discussions may be used to stimulate ideas that are technical advances beyond the current situation or outside the current paradigms associated with the project. Beyond this, the strength of the vision is to provide the project team with a clear and common direction to their efforts. To be effective, the vision should be established with the customer, reviewed with team experts, and circulated among the team members in a high profile manner. The vision must be something that the team can become excited about and support, but at the same time, be realistic and feasible within the constraints and available resources of the organization.

3.3.4 External Factors

Time is taken during the goals analysis for explicit discussions of the external political factors affecting the project and the likely implications to the development. Typically, these discussions will include such topics as NASA Headquarters sponsorship, funding trends, related administrative policy, impact on the public, and existing or likely competitors. The purpose is to explicitly address the unwritten rules which will affect the project and to develop plans for clarifying, qualifying, and reducing any related risk from external constraints. Uncertain political situations should be a part of the overall project risk assessment as they may have a greater impact than the technical risks. Consideration should be given to other groups affected by fulfillment of the top goal for the purpose of developing a constituency or advocacy group. The political advantages of pursuing the project goal as stated, as well as the drawbacks, should be considered. The attitudes and opinions of those with decision-making authority over the project must be carefully considered. If opposition appears unmanageable, a modification of the goals may be in order.

3.3.5 Goals Tree

The culmination of the goals analysis is the development of the hierarchy of project goals, or the goals tree. This is a graphic depiction of the goals structure, with appearance similar to an organization chart, which shows the top-level goal and its relationship to more specific supporting goals. The top-level goal, as previously developed, is displayed at the top of the tree, with supporting goals below. Those in a supporting level are referred to as sub-goals which "will support" the accomplishment of the goal at the next higher level. Each sub-goal is further refined to a subsequent level which "will support" its fulfillment. A branch is terminated when the sub-goal meets the qualifications for requirements (see Section 3.4) which assures that the sub-goal may be indisputably judged as met (or failed). As an illustration, a sub-goal which was worded "to provide for minimum delay in data transfer" would be insufficiently defined to warrant termination of that branch of the tree. This sub-goal would require further breakdown and may terminate the branch with a statement such as "to transfer 230 kilobytes of information to the ground station within 15 seconds." It should be noted that even at the terminating nodes of the tree, the requirements should describe precisely what is to be accomplished without describing how the accomplishments are to be made. "To be determined" (TBD) is a useful placeholder for numerical entries, but should be replaced with target values as soon as possible.

One branch from a goals tree is shown in Figure 3.1, "Sample Branch from Typical Goals Tree." The progressive decomposition from the top level goal through sub-goals to the specific science functional requirements at the lowest level is clearly illustrated. The shaded boxes are the "leaf" terminations of the branch.

The significance of this process is that system requirements are tied directly to the top-level goal of the project.

Goals Hierarchy for LIDAR at LaRC

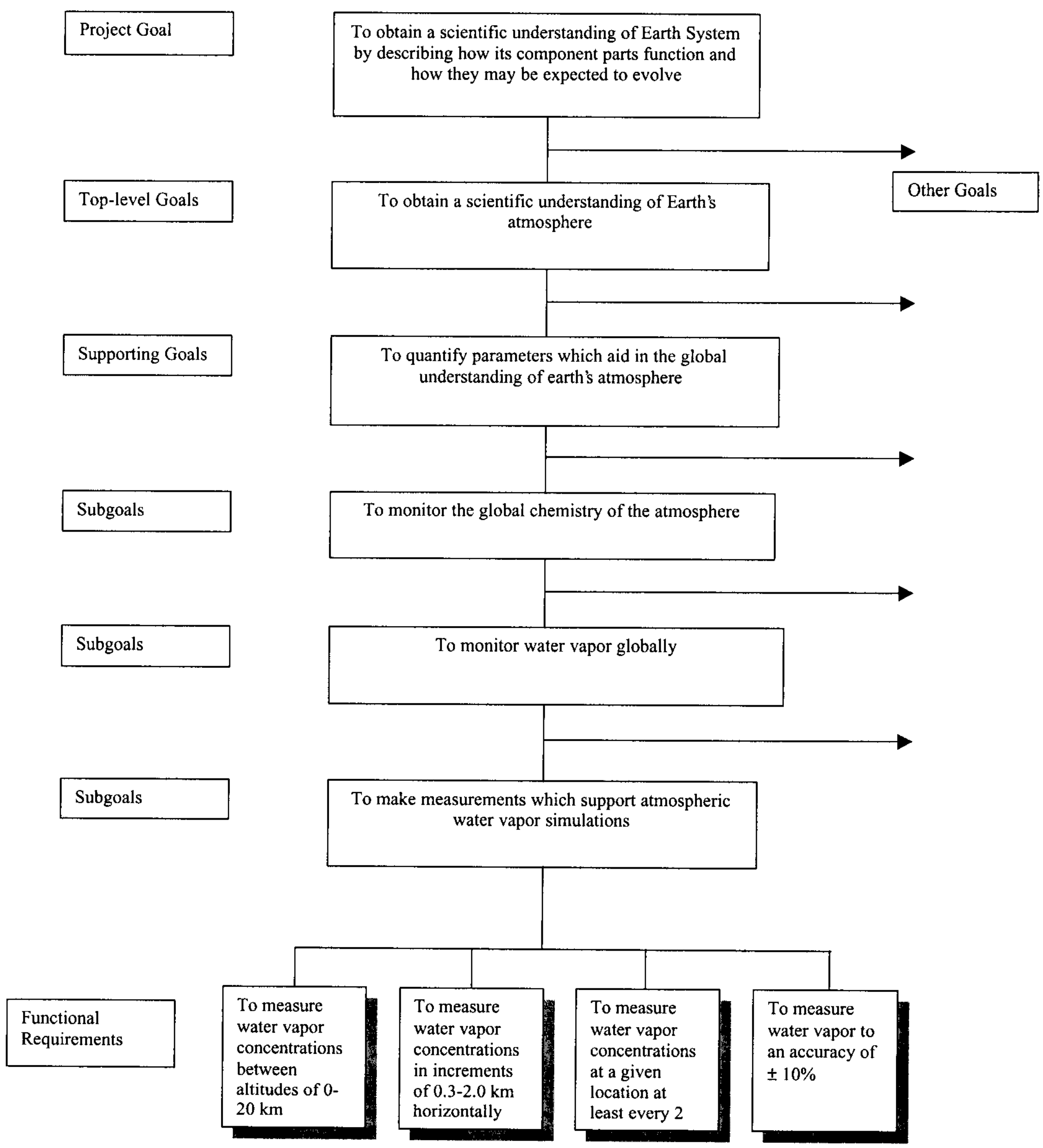

Figure 3.1 – Sample Branch from Typical Goals Tree.

3.3.6 Validation of Goals

To this point, the goals analysis has been an outscoping exercise that has solicited input without criticism. Next, the work is reviewed and scrutinized for error, completeness, and consistency. Where criticism was previously discouraged, it is now invited. Compatibility with the Langley Research Center (LaRC) vision and the extent to which the project fulfills a NASA Headquarters strategic plan milestone should be assessed. After this review, the goals analysis process is iterated so that any points which surfaced during the subsequent application may be fed back into the appropriate steps. The first iteration should be a rough cut to get the pertinent issues out into the open. This may typically be accomplished in a day of meetings. The information must then be organized by the systems engineer for presentation to the team at the next meeting.

3.3.7 Summary

The product of this step will be the Goals Analysis Document which will initially be issued in a preliminary form and gradually refined as the process is repeated in the later phases of the project life cycle. The goals document will form the basis for all subsequent project activities including requirements flow down, verification, and validation standards.

3.4 SYSTEMS REQUIREMENTS AND CONSTRAINTS

This step of the Systems Analysis and Design Procedure defines requirements and constraints at each phase of the life cycle leading to progressively more detailed definition of functional requirements for the hardware and software architectural hierarchies. The input for this activity is the terminating branches of the goals tree. Essentially, these functional requirements, or objectives, define what must be done to achieve project goals. They provide a quantitative description of measurable entities. Requirements may be considered as one of two types: constraints or performance measures. Constraints limit the set of possible options by establishing system boundaries. Performance measures are the variables used for system and subsystem trade-off analysis. An example of a constraint would be spacecraft launch weight which must be kept below a certain mass for a particular orbital condition. Performance measures are discussed in Section 3.5. All types of requirements must meet certain necessary qualifications. By definition, requirements must include, as a minimum, the properties described in Sections 3.4.1 through 3.4.5.

3.4.1 Quantifiability

Some numeric value must be assignable to the requirement. The quantifiable nature of a requirement is the attribute which allows the determination of success. Requirements such as "minimum downtime" or "maximum data rate" cannot be verified without further definition. The requirements must be put into the form of a numeric value such as ". . .

downtime of less than 100 seconds per 30 days" or ". . . data rate of greater than or equal to 5 kilobits per second." Just as in cost and schedule requirements, there must be some quantitative measure associated with each parameter.

3.4.2 Objectivity vs. Subjectivity

This means that the definition of terms is the same for everyone. An example of a quantifiable, but subjective measure, may be "the number of acceptable data points must be greater than . . ." The term "acceptable" is not an objective standard.

3.4.3 Meaningfulness

This qualifier is added to assure that measures directly address the goals of the project, as opposed to choosing criteria simply because they may be easily counted or for personal reasons. Deductive requirement development through the goals tree is the best method of excluding spurious requirements.

3.4.4 Absolute Scale

Requirements should not be measured relative to some other parameter that may vary as this leads to confusion in the assessment of the satisfaction of requirements. An example of a relativistic requirement may be ". . . shall be degraded by less than 20 percent due to solar activity."

3.4.5 Verifiability

This is an extremely important requirement characteristic since any requirement that cannot be verified by some method should be discarded as meaningless. Note that typical methods of verification include analysis, as well as inspection and test.

3.4.6 Summary

The product of this effort is the Systems Requirements Document and, ultimately, the requirements database for the project. The Systems Requirements Document will include all of the systems requirements including the instrument, mission, spacecraft, and operations requirements. The requirements definition will become progressively more refined during each project phase and these systems requirements will eventually be flowed down through the hardware and software hierarchy to the appropriate levels.

3.5 PERFORMANCE MEASURES

Performance measures are defined during the goals analysis to provide criteria for subsequent trade studies of alternate system configurations. A subset of the project objectives, performance measures, are variable parameters used to provide a quantitative description of the system's ability to satisfy functional goals. The parameters will usually have a limit on the minimum acceptable level of performance, but achievement beyond that limit is of significant value. For example, while an instrument may have a minimum performance accuracy of 90 percent, an option providing 95 percent accuracy might be more desirable and worthy of additional cost. This is in contrast to constraint type objectives such as system volume, in which no real advantage is realized by performing better than the required value. The performance measures are typically ranked to provide an indication of their relative importance. The results of this process are compiled in the Performance Measures Statement.

3.6 SYSTEMS CONCEPTS

Ideally, any and all potential system concepts should be considered in order to describe the entire range of possible solutions. However, to prevent being overcome by a large number of possibilities, classes of concepts have sufficient definition for initial studies. As analyses continue, clear front runners emerge in each of the feasible classes, which are studied in greater detail.

The most widely used technique for identifying options is brainstorming, but more all inclusive techniques, such as combinatorics, may be used to generate alternatives. A combining approach first breaks the general concept into subsystems, elements, or functional segments, and then considers the possibilities on each decomposed piece. The various options are then recombined to form specific system concepts. This may often bring to light concepts that were not previously considered. The output of this activity is a set of candidates that are described with only the essential details. A quick review will likely remove some options as unfeasible, or obviously incapable of satisfying constraints. The surviving alternate concepts then progress to the next step in the process, in which candidate performance is predicted. As the project progresses, the list of alternate concepts is reduced until the baseline system concept is selected in Phase B. As the system is decomposed into smaller pieces, the process is repeated for alternative element, subsystem, assembly, and subassembly evaluation and selection.

3.7 CONCEPTS ANALYSIS

Once the set of possible concepts has been developed, performance of each must be projected as an input to decisions. The previously established performance measures define precisely what performance must be projected. The concept characterizations are the "bottom-up" or inductive complement to the deductive work accomplished in the goals analysis. Estimates are sought for each feasible solution (one which meets constraints) on the value for each performance measure. A closer look will likely identify additional concepts which cannot meet the minimum requirements.

Mathematical modeling and computer simulation are very useful at this stage, especially for new developments about which there is no existing performance information. Probability distributions may be even more useful than point estimates of variables since they also describe the uncertainty associated with each projection. Known values are recorded at their specific values. Note that cost and schedule projections of the alternative concepts are always required as a minimum. This analysis of concepts and performance measure projection allows for a structured approach to determining the overall best solution, through use of decision analysis.

The product of this effort is the Alternate Concepts Analysis Document which provides a comparison of each of the alternate systems. The alternate concepts are eventually reduced to the baseline system concept during Phase B. More detailed analysis of concepts is typically done as the system is decomposed down to the subsystem, assembly, and lower levels.

3.8 CONCEPTS RANKING

This step is the culmination of the procedure in which the set of systems concepts is ranked through decision analysis for the purpose of indicating their overall attractiveness. The inputs to decision analysis are the performance measures and their relative weights, the set of feasible alternative concepts options, and the estimations for each option on each performance measure. The output is a list of surviving solutions in numerically ranked order.

To accomplish the ranking, the performance measure estimates that each concept is first normalized to a value between 0 and 1. The normalized numbers are combined with performance measure weights to produce an overall score for each concept. The advantage of a structured decision approach is that a very difficult, multiple attribute decision may be broken down into a number of more simple, single factor judgments. In a more sophisticated implementation, it is possible to express the variation in a decision maker's utility over the range of possible values of a performance measure. Also, a number of models exist for the overall combination of the performance measure weights and concept values. Regardless of the approach taken, the project will be structured in a fashion to allow for easy analysis of the solutions.

Since there is subjectivity in the weights of performance measures, a variation in these values will reveal the impact on the order of preference of the candidate concepts. A robust solution, one which is relatively insensitive to variations in the performance measure weights, is preferred. Sensitivity analysis may be used to graphically display the effects of variation in performance measure weights. Example decision analyses are shown in Appendix D.

3.9 SYSTEMS DEVELOPMENT

In the formulation phases, concepts are designed and breadboard or brassboard hardware and prototype software are developed in support of risk reduction activities. During the implementation phases, the conceptual layouts of Pre-Phase A, Phase A, and Phase B are formally designed, fabricated (or procured), and integrated and tested. The sequence of development may include prototype or engineering model hardware with supporting software. In all phases the development effort may be subject to changes in schedule, changes in funding profile, or technical performance required. In addition, during the verification process, it may be discovered that the technical performance desired is unachievable under time and money constraints. Renegotiations among the project, funding bodies, and customer will be necessary in these cases to decide on an altered approach; for example, seek additional funding, allow schedule slip, or accept lesser technical performance. Frequent team meetings and status reports from subsystem managers are crucial to identify problems before they are unachievable within project constraints. Likewise, extreme attention must be paid to the definition and maintenance of interfaces among subsystems to allow for easy integration. The systems engineer must be continually alert to the occurrence of increased risk in the system. The sequence of development is usually driven by external constraints or risk mitigation.

3.9.1 Technical Risk Management

Beginning in Pre-Phase A of the project, risk aspects of the development must be identified, characterized, and mitigated to an acceptable level. The systems engineer is responsible for reduction of technological risk and for development of the Risk Reduction Plan. For LaRC, this risk is inherent in most projects because of the research orientation and lack of previously demonstrated performance. Thus, a systematic approach must be undertaken to productively reduce the project technological uncertainties. Risk (or expected outcome) is defined as the severity of the occurrence of an event, multiplied by the probability of that event or:

$$\text{Risk(Event A)} = p(A) * \text{Cost}(A)$$

where $p(A)$ is a value between 0 and 1 and $\text{Cost}(A)$ is usually described in dollars. This returns the value of Risk(Event A) in expected cost to the project. Thus, it may be seen that catastrophic hazards may (and usually are) acceptable if the probability of the event is sufficiently low.

Conversely, a relatively innocuous event may require risk reduction if the probability of the event is high. The approach to risk reduction consists of three steps: identification, quantification, and reduction as outlined below.

3.9.1.1 Risk Identification

The best methods of risk identification include brainstorming (with experts in the field of endeavor) and review of lessons learned. As in other inductive techniques, this exercise should begin without criticism in order to explore all possible sources of risk. Subsequent quantification will quickly identify those risks worth tracking. A list of technical risk areas should be compiled and maintained for the life of the project. As risk areas are reduced in value, they may be removed from the list of those currently under review, but should be maintained for the project audit trail, and to serve as lessons learned for future projects.

3.9.1.2 Risk Quantification

This step is necessary in order to determine which of the identified risks require action. Expected costs may be determined as above, or more sophisticated calculation techniques may be utilized if the data is obtainable. A slightly better approach is to estimate the most pessimistic, most optimistic, and most likely costs, should the hazard occur. These values may be weighted and added as follows:

Risk(Event A) = [2/3 * Cost(A likely)]+
[1/6 * Cost(A optimistic)]
[1/6 * Cost(A pessimistic)]

This approach is sufficient for most applications.

If probability distributions are available (for example exponential, Weibull, and so forth) then the functions may be combined as:

Risk(Event A) = †0* xf(x)dx

where:

x = Cost of Event A
f(x) = Probability of Event A

Like the two other methods, this equation will return the expected cost of Risk (Event A).

A qualitative approach to risk assessment may also be taken. In this case, the judgment of experts is used to prioritize risk areas based on past experience. This identifies a rank order of risk events needing attention, which may be subjected to risk reduction and tracked to closure.

3.9.1.3 Risk Reduction

After the risk events are quantified, they must be examined to assess their acceptability. For those risks considered too severe, structured steps must be taken to reduce or remove the impact of the event. It is important to note that in practical cases, there should not be more efforts expended on risk reduction than the expected value of the risk itself. However, if the risk could lead to total system failure, the cost to reduce the risk may be justifiable up to the total cost of the system, given the probabilities are high. In these cases, additional research is usually indicated to mature the concept before the project development begins.

The two basic strategies for reduction of risk are to decrease the probability or decrease the severity of the risky event. The following paragraphs illustrate approaches to risk reduction from both of these approaches:

*Decrease the Probability

If the possibility exists to delete that portion of the system responsible for a given risk (in favor of a less risky alternative), this should be considered first. Obviously, in many cases this will not be feasible. For example, the limited life span of detectors cannot be removed from system risk by eliminating the use of detectors, since they are fundamental to system operation. Rather, the approach may be to develop or qualify certain long-life detectors to satisfy system requirements. Another approach may be to strengthen the integrity of the weak points of a design by utilizing higher reliability components. In the event that alterations to the system are not feasible, some additional control may be possible through the introduction of procedures that are implemented to heighten confidence in the integrity of the system.

*Decrease the Severity

An alternative to the above is modification of the system design to include backup systems. This should lead to a so-called "graceful degradation" in which a component failure will lead to a decrease in overall performance, but will not result in a total system failure. An example may be seen in satellite communications, where a failure of the primary system would result in operation of a backup transponder. The backup may move much less data, but would still be sufficient to transmit the most critical information. Another possibility may be to introduce warning systems that alert the system operator of impending problems. This approach is valid only if the mechanism and time exists to alleviate the problem before a catastrophic failure occurs.

3.9.2 Other Risk Categories

The risk related to items within the boundary of the systems under development is addressed above. However, there are other types of risk that are imposed on the project by external concerns. As a minimum, the systems engineer must be concerned with safety risk and schedule risk. Safety risk is concerned with control of events that pose a hazard to entities outside the system boundary; most notably personnel and ancillary equipment. For LaRC projects, the reduction of safety risk will be overseen by a product assurance engineer who is responsible for the imposition of safety oriented requirements. Typically, safety risk is mitigated through regulations or constraints imposed by the organization responsible for the launch of the system. For many LaRC flight projects, these requirements are levied by the National Space Transportation System (NSTS) safety organizations at Johnson Space Center (JSC) and Kennedy Space Center (KSC). In a similar fashion, safety requirements are established by the parties responsible for alternate launch vehicles and launch facilities.

Schedule risk must be dealt with directly by the systems engineer. Schedule risk may be associated with development time for an item or technology on the critical path, with procurement time for a critical item, or with personnel scheduling problems. This risk is described in expected system delivery slip in units of time. High schedule risk events are typically addressed through contingency planning and the development of technologies which are parallel to the baseline approach.

3.10 REVIEW, VERIFICATION, AND VALIDATION

As outlined in Section 3.4, requirements must have some associated method of verification. Typically, the term verification relates to system specific hardware and software requirements. In contrast, validation is concerned with assuring that the system meets the needs of the customer. Thus, system validation truly occurs only in flight. However, while complete system testing prior to flight will not prove the capability to meet customer needs, integrated system verification can certainly identify the inability to do so.

In the system life cycle, verification usually begins in risk reduction efforts. Certain portions of the conceptual system are built or modeled in an effort to verify their ability to satisfy certain hardware or software requirements. Life testing may be indicated in the early project stages to establish expected life of an unproven design. Beyond risk reduction, verification occurs as the first portions of the engineering model are received or assembled. This usually assesses the ability of a subsystem to meet its performance as required in its anticipated environment. This testing will continue at higher levels as larger portions of the system are assembled. Ultimately, this process leads to system level verification and validation in flight.

Review is a constant element of the systems engineering process for the purpose of error control and improvement. The steps of the process are performed as applicable in each project phase and iterated to assure the accuracy of the products. This internal iteration occurs on a daily basis within the project. The formal, external reviews occur as control gates between phases.

3.11 DECISION POINT

The culmination of the Systems Analysis and Design Procedure for each phase is the decision made on the basis of the phase study products. In an LaRC development, this decision will occur at the end of each phase of project formulation when a review is convened to assess the readiness of the project to proceed. This review process allows the customer to assess the state of the project and make decisions concerning future directions. While the result of convening a control gate review may be to proceed without condition or to cancel the effort, the outcome will typically fall somewhere in between. Additional work, as required by control gate action items, will usually be required before proceeding to the next phase. Regardless of the final outcome, control gates should be treated as true decision points relative to the future of the project, and not simply an exercise for the project team.

Chapter 4

PROJECT LIFE CYCLE

SUMMARY

This Chapter describes the NASA project life cycle that uses a phased approach, defined goals, and measurable milestones to lead to the accomplishment of overall project goals.

4.1 INTRODUCTION

The NASA project life cycle is a phased approach that organizes project activities into a logical sequence of steps from initiation through completion. The project life cycle concept is a systematic method to organize a major effort into a series of progressive steps with well defined phased goals and measurable milestones leading to the satisfaction of overall goals of the project.

A summary of the project phases, objectives, and control gates are shown in Figure 4.1. The activities in Pre-Phase A, Phase A, and Phase B are termed the formulation phases of the project since the emphasis is on requirements analysis, project planning, concept definition, and feasibility demonstration. Phase C, Phase D, and Phase E are termed the implementation phases because operational hardware and software are designed, fabricated, integrated, and become operational. Typical phase durations are shown in the Project Phase column of Figure 4.1.

A time based schedule chart is shown in Figure 4.2, "Space Flight Project Life Cycle." Two charts are shown representing both a minor, short duration project and a major, long term effort for a sophisticated flight system. The chart displays typical phase durations, along with the associated timing of project reviews. Figure 4.2 shows the close relationship between the initiation of a major project and the NASA related funding activities shown in the Funding Activities Plans column. The funding activities require lead time; for example, Phase B funding plans must occur during Pre-Phase A activities.

4.1.1 Funding

Although the initial stages of concept development may proceed independently, an evolving project must very quickly become synchronized with the NASA budget cycle. If the project is part of a larger program, the project initiation may be keyed by response to an inter-center Announcement of Opportunity (AO). If so, the AO will usually describe the sequence of preliminary studies leading up to a project proposal which will be the basis of selection for a new start activity in a future fiscal year. When a project is accepted and funded to begin Phase C, it is commonly termed a "new start" or full authority to proceed (ATP). If the project is an independent activity, then the New Start Proposal may be incorporated in the Center Program Operating Plan (POP) for the new start year (NSY). The budget process may extend over a 3-year period leading up to the NSY.

PROJECT PHASE	OBJECTIVE	CONTROL GATES
Pre-Phase A: Advanced Studies (1 to 3 months)	Preliminary requirements and concepts analysis	Sponsoring Group Director's Review (SGDR)
Phase A: Preliminary Analysis (2 to 6 months)	Requirements definition and conceptual trade studies	Preliminary Systems Requirements Review (PSRR) SEIRC Review* Group Directors' Review (GDR) LaRC Center Director's Review NASA Headquarters' Review
Phase B: Definition (4 to 18 months)	(B1) Concept definition and preliminary design (B2) Source selection process (if required)	Systems Requirements Review (SRR)* [Conceptual Design Review (CoDR)] Software Concept Review (SCR) Software Requirements Review (SRR) Preliminary Design Review (PDR)* Software Preliminary Design Review (SPDR) Non-Advocate Review (NAR) Source Evaluation Board (SEB) Review
Phase C: Design (9 to 24 months)	Final design and engineering development	Critical Design Review (CDR)* Software Critical Design Review (SCDR)

Phase D: Development (9 to 24 months)	Fabrication, integration, test, and evaluation	Test Readiness Review (TRR) Software Test Readiness Review (STRR) Software Acceptance Review (SAR) System Acceptance Review (SAR)* [Pre-Shipment Readiness Review (PSRR)]
Phase E: Operations (3 to 12 months or longer)	Preflight and flight mission operations and disposal	Operational Readiness Review (ORR) Flight Readiness Review (FRR)* [Launch Readiness Review (LRR)] [Preflight Review] Operational Acceptance Review (OAR) [Post-flight Review] Lessons Learned Review*

*Formal reviews required per LAPD 7120.1.

Figure 4.1 - NASA Project Life Cycle.

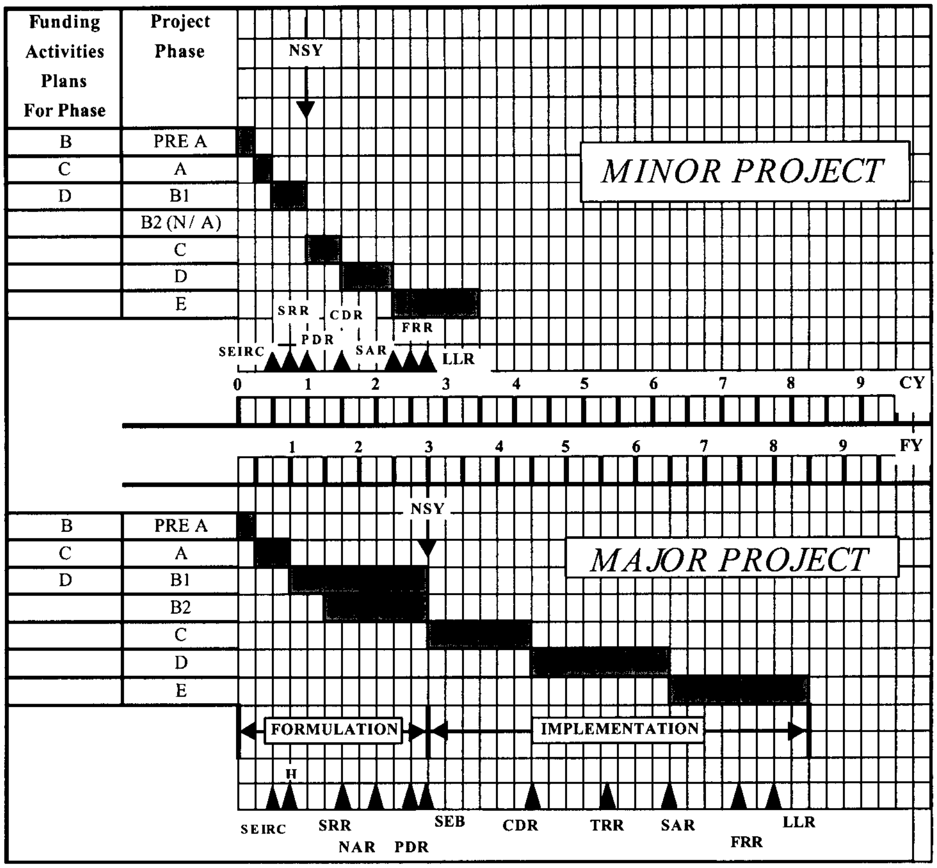

Figure 4.2 – Space Flight Project Life cycle

If the program is considered a major system acquisition as defined in NPG 7120.5A, NASA Program and Project Management Processes and Requirements, then the first key activity will be the submittal of a Mission Need Statement (MNS) approximately 3 years prior to the NSY. A smaller program may be initiated by means of a Project Initiation Agreement (PIA) between NASA Headquarters and LaRC. If approved, the program would typically be included in the POP for the NSY. The budget inputs are normally solicited in the spring of the second year prior to the NSY. The Center inputs are submitted to the NASA Program Office in the summer and the NASA budget recommendation is submitted to the Office of Management and Budget (OMB) in September about 1 year in advance of the NSY. The final budget is prepared by OMB and approved by the President for submittal to Congress in January. The Congressional budget process, including action on appropriation and revenue measures, begins 9 months prior to the fiscal year. The budget is scheduled for enactment and approval by October 1 of the NSY. Initial appropriated funds are apportioned to the Agency within a few weeks after the start of the fiscal year and the balance are received incrementally during the budget year.

Funding for the project formulation phases and for smaller projects is usually arranged on an ad hoc basis. The effort expended during Pre-Phase A is relatively minimal and personnel hours are usually funded by an ongoing, funded, research and development program. Similarly, Phase A activities are usually funded in-house from the related Research and Technology Operating Plan (RTOP) of the research and technology program or directly from the cognizant Program Associate Administrator (PAA) at NASA Headquarters.

Funding for Phase B activities must usually be obtained from a Program Office at NASA Headquarters. Phase B focuses the project planning and one of the outputs of Phase B is the Project Plan that will supersede the earlier PIA. One of the major challenges of the study/project team is to obtain and administer funding for the formulation phases so that the project goals can be addressed on a timely basis and the young project organization can be supported and held together until Phase B and/or Phase C funding is received.

It can be seen from the above general discussion that the project life cycle, for a large project, is driven by the NASA budget process. The Pre-Phase A effort must be completed as expeditiously as possible (usually within a 1- to 3-month period) to substantiate the beginning of the more detailed Phase A study. Phase A and Phase B for "complex" efforts must usually span the 2- to 3-year interval until the NSY. For example, Phase A for a complex project might last 6 months to a year from start to completion of the LaRC Center Director's and NASA Headquarters' Reviews. Phase B for concept definition might last a year until completion of the Systems Requirements Review (SRR), Non-Advocate Review (NAR), and the Preliminary Design Review (PDR). Another year might be required if a major acquisition activity with a Source Evaluation Board (SEB) is required.

4.1.2 Concurrent Engineering

The importance of a well-integrated and informed team cannot be overstressed. Principles of concurrent engineering should be practiced from the onset of a project. Practically speaking, this means that the phase teams should be represented by members from all related functional fields such as hardware design, software development, manufacturing, project controls, testing, and so forth. The team should meet regularly for status updates and the exchange of ideas and viewpoints. This application of concurrent engineering is especially critical among subsystem managers with shared interfaces. One responsibility of the systems engineer is to ensure that these interactions occur among members of the project team.

4.1.3 Systems Engineering Analysis and Design in the Life Cycle

The application of systems engineering analysis in the project life cycle occurs at several levels. In the formulation phases of the life cycle the systems analysis process comprises the bulk of systems engineering activities. Early in the life cycle, systems engineering analysis is applied on the system level to analyze various approaches, and thus, support the selection of a baseline concept. The application of the systems engineering process, and related consideration of multiple conceptual options, is in sharp contrast to the "point design" approach. A point design occurs when the project team puts all efforts into a single alternative from the inception of the new project. While this approach may have been valid in the past, the increasing sophistication of advanced systems has rendered all but the simplest developments too complex for immediate design.

In formulation, the Systems Analysis and Design Procedure is guided by the systems engineer toward satisfaction of the principle customer goals. Thus, performance measures will be relative to overall project goals and typically related to the systems ability to perform a desired function. Examples of these functional performance measures may be: measurement accuracy, resolution, range, or data transfer rate. As a minimum, systems engineering analysis will be performed at this level for all projects.

Subsystem managers may wish to take advantage of systems engineering analysis for selection of subsystem configuration. In this case, the subsystem manager is usually designing within the engineering constraints and budgets allocated by the systems engineer. Weight, power, volume, heat flux, alignment, and other such parameters may be viewed as engineering constraints on a subsystem development. Variable performance measures for the subsystem engineering analysis, commonly referred to as trade studies, will likely take on other forms such as throughput, processing time, or material strength.

The Systems Analysis and Design Procedure is valid to the lowest level of hardware piece part selection and software unit development, but the value of the application will reach a point of diminishing return. At some point in the detail design, solutions become readily available based on experience, and details are captured in the form of drawings and specifications.

As the project progresses into the latter portion of Phase B and into Phase C, systems engineering analysis is used by subsystem managers, but other duties, such as hardware/software integration and verification, begin to dominate the attention of the systems engineer. These activities are outlined in detail in the following sections of Chapter 4 which present a general overview of the NASA project life cycle. A more detailed treatment with specific activities and products is given in Chapter 6 when the systems engineering process and the project life cycle are combined.

Each of the following phase descriptions contains a paragraph on team composition. The actual makeup of a phase team will be determined by the Project or Study Manager. The information presented here is meant to be indicative of the functional disciplines that typically participate in a given phase.

4.2 PRE-PHASE A - PRELIMINARY REQUIREMENTS AND CONCEPTS ANALYSIS

4.2.1 Pre-Phase A Team Composition

The Pre-Phase A team is established at the beginning of the phase and should, as a minimum, include the following functional representation:

* Science/Technology
* Study management
* Systems engineering
* Experts in the major hardware fields related to the project (that is, lasers, electronics, spacecraft, and so forth)

While the core team will probably consist of approximately seven members, contact with other experts, including software engineers, will be required on an as-needed basis for conceptual feasibility, cost, and schedule assessments.

One item of note in the above list is the presence of a Study Manager in place of a Project Manager. This distinction is made to emphasize the fact that the project is not yet mature enough to assume project status. A Project Manager is appointed in Phase A, and will ideally stay in that position for the balance of the project life cycle.

4.2.2 Pre-Phase A Implementation

The purpose of the Pre-Phase A study is to quickly assess the feasibility of a proposed project and develop a Go/No Go recommendation for Phase A. This first requires a thorough understanding of what is desired by the customer of the study. Given this information, concepts may be evaluated to determine what approaches, if any, are feasible. A critical question for resolution is the appropriateness of evolving the study into a project proposal. Sufficiency of work completed will be reviewed by the LaRC sponsoring Group Director at the end of Pre-Phase A to decide if progression into Phase A is warranted.

The advantage of applying a disciplined systems engineering process to the Pre-Phase A study is that the goals of the proposed project can be quickly focused, requirements defined, and the various system concepts can be developed and evaluated within a span of 1-3 months. Thus, promising ideas can be rapidly identified for further study to shorten the overall life cycle time from project initiation to completion.

4.3 PHASE A - REQUIREMENTS DEFINITION AND CONCEPTUAL TRADE STUDIES

4.3.1 Phase A Team Composition

The Phase A team may be somewhat larger and more specialized than the team from Pre-Phase A. Attempts should be made to retain those team members from Pre-Phase A with the best understanding of the concepts which were selected for further study. Additional support will be required from software engineers and product assurance support should begin. The following functional representation is typical:

* Science/Technology
* Project management
* Systems engineering
* Experts in the major hardware fields related to the concept options surviving Pre-Phase A
* Systems analysts (thermal, structural, control, others as required)
* Software development
* Product assurance engineering (may be delayed until Phase B at discretion of the Project Manager)

As in Pre-Phase A, contact with other experts will be required in support of concept design and analysis. The size of the Phase A team will be roughly the same as the Pre-Phase A, with extra disciplines added as required. Preliminary analysis of system thermal, structural, and control characteristics may be required. These team members are likely candidates to stay with the project into Phase B. A significant difference in the scope of the Phase A effort versus Pre-Phase A is the depth and duration of the analysis.

4.3.2 Phase A Implementation

The emphasis of Phase A is to do a more detailed definition of mission needs and requirements and to conduct more detailed trade studies and analyses of the best, feasible alternative system concepts identified in Pre-Phase A. To accomplish this, a Project Manager is appointed and the project staff is augmented for the 2- to 6-month effort. Most of the study will be done in-house, but contractor support may be utilized in specialized areas and other industry inputs may be sought to assure that the study results will be balanced.

The goal of Phase A is to establish which of the feasible system concepts under consideration are the most preferred. These concepts may be ranked in accordance with the various performance measures, but there is no action to select a single baseline conceptual design at this point. The results of Phase A studies are presented in a preliminary New Start Proposal; and when appropriate, a Mission Needs Statement to NASA Headquarters which can be used to justify funding for a Phase B effort.

Participation by technical hardware and software team members is required during the formulation phases to assure the proper allocation of system requirements to lower levels of assembly. Early partitioning of system functional requirements to software and hardware, if feasible, will greatly support timely system development. However, all pertinent system level studies should be completed before requirements are allocated. Actual software development activities will frequently be tied to risk reduction at this stage of the project. Often, software prototype models are required to serve as a simulator for a portion of the system that has yet to be developed. The purpose of the simulator is to interface with preliminary hardware which is built to increase confidence in the design approach. The results of these activities are maintained in the Systems Requirements Document and the Requirements Data Base.

Control gates during Phase A include a Preliminary Systems Requirements Review (PSRR) and a review by the LaRC Space-flight Experiment Initiatives Review Committee (SEIRC). The function of this committee is to review and critique proposed space flight experiments prior to presentation to upper management. These reviews are followed by briefings to the LaRC Group Directors and to the Center Director. Finally, the Phase B plan is presented to NASA Headquarters with a proposed PIA. The PIA outlines the new project's management and technical interfaces, schedules, resource estimates, and other ground rules and becomes the initial agreement between LaRC and the NASA Headquarters Program Associate Administrator (PAA) who will sponsor the project. At the completion of Phase A, a preliminary set of requirements and a rank ordered set of feasible system concepts are available. This is in contrast to the majority of past efforts which focused on a single concept early in the project formulation.

4.4 PHASE B - DEFINITION

4.4.1 Phase B Team Composition

In Phase B, the size of the project team will increase somewhat. The Project Manager will be responsible for selecting subsystem managers who will have responsibility for functional hardware and software segments of the system and for establishing the organizational structure of the project. After the baseline concept is selected, additional support will be required in new functional areas such as testing and quality control. A typical Phase B team consists of:

* Science/Technology
* Project management
* Systems engineering
* Subsystem management
* Experts in the major hardware fields related to the concept options surviving Phase A
* Systems analysts (thermal, structural, control, others as required)
* Software engineering
* Product assurance engineering
* Testing
* Project controls (costing, scheduling, and configuration management)
* Quality control
* Electronics manufacturing
* Hardware manufacturing

The team additions shown above will support risk reduction efforts, project planning, and the implementation of concurrent engineering.

4.4.2 Phase B(1) - Concept Definition and Preliminary Design

Phase B(1) is primarily concerned with the task of concept definition in order to establish the baseline system design and also the optional source selection process which will be discussed separately. The Phase B team will be staffed with personnel who are experienced in conceptual design studies and analyses.

The greatly expanded efforts in Phase B will result in a related increase in the size of the project team. Thus, initial Phase B activities will begin with bringing the new team members "up to speed." This emphasizes one of the important advantages of the structured systems engineering process. Since the process is cyclical in nature, new team members have the opportunity to review, verify, and refine work accomplished previously in the project.

During Phase B, the project goals and requirements are reviewed and the Goals Analysis Document and Systems Requirements Document are finalized. A mature Performance Measures Statement is defined and more detailed verification and validation plans are developed. If required, the final systems engineering management plan covering in-house systems engineering activities should be issued.

At the beginning of Phase B, the candidate system concepts are reviewed and further tradeoff studies are performed, if necessary, leading to selection of one approach as the baseline concept for detailed study. This activity is required because of the significant increase in size and diversity of the project team. Detailed systems design studies are performed on the overall system and on the technical discipline subsystems such as electrical, thermal control, structural, electronics, software, and so forth. Of particular importance here is the development of the system architecture model that encompasses both hardware and software functions. This mathematical or systems simulation model must demonstrate that the predicted end-to-end system performance will meet the overall system requirements and satisfy the user needs and goals. This model or prototype should closely approximate the user interfaces but the internal hardware and software aspects will be very preliminary. This systems simulation model will be upgraded and maintained throughout the life of the project. The results of these studies are presented in the Baseline Design Concept Package and become part of the New Start Proposal.

Risk analyses and risk reduction activities are addressed to identify areas requiring further technology development. As further refinements are made to the conceptual design, the system Work Breakdown Structure (WBS) is expanded and the life cycle cost analysis is updated. The overall study effort is finalized and becomes more focused as the Systems Requirements Review (SRR) package is prepared and presented to an LaRC review team. This will address both hardware and software requirements as they have been allocated from the system level requirements.

One aspect of development which becomes important in Phase B is the allocation of system resources to subsystems. Such parameters as weight, power, volume, alignment tolerances, and heating/cooling capacities are vital commodities for space hardware and must be carefully controlled. The total amount available, or system budget, of a given resource must be divided among subsystems, carefully tracked, and reallocated as necessary. Each subsystem manager is responsible for assuring that their portion of the system stays within the guidelines provided by the systems engineer. Should it become apparent that an allocated budget is insufficient, as it usually will at some point, the systems engineer must either pull resources from another sub-system, determine the impact of decreased performance of the subsystem in question, or draw from resources held in reserve. Since the latter of these options is frequently the most attractive, it is very important for the systems engineer to hold a significant portion of system resources in reserve, especially in the earlier stages of development. (See Chapter 7, Section 7.5.)

Action items from the SRR are addressed and further studies are accomplished to enhance the baseline conceptual design. Subsequently, the team will present a formal Non-Advocate Review before a committee at NASA Headquarters or at a host field installation program office to obtain funding for Phase C and Phase D. At this time, the proposed Project Plan will also be reviewed. Release of funds for the implementation phases (Phase C and Phase D) is dependent upon project approval and authorization by NASA Headquarters and subsequent allotment authority for obligation of funds at the Center level.

The Preliminary Design Review (PDR) is also completed during Phase B. The PDR package is prepared and presented when the hardware and software designs are about 70 percent complete. Any PDR action items must be addressed prior to start of the final design phase. Following this review, the baseline concept will become the preliminary As-Designed Project Baseline under formal change control and Phase B(1) activities will be completed.

4.4.3 Phase B(2) - Source Selection Process

This phase is necessary if major portions of the system are to be contracted to industry. In the event of a major procurement, the project schedule may dictate that the acquisition process actually begins in Phase A or early in Phase B with the preparation of a Statement of Work (SOW) and preliminary technical interchanges with possible sources. It is also important that acquisition personnel be an active part of the Phase B team so that they can be informed about the scope and technical content of the proposed acquisition and have sufficient time for planning the procurement.

LaRC acquisition procedures are summarized in NPD 5101.32, Procurement. The project team or a group designated by the Project Manager will be responsible for preparing a procurement package technical specification or an SOW for the procurement. This group should contain representatives from hardware and software disciplines. When the contract is awarded, a project team member will be delegated by the Contracting Officer as the Contracting Officer's Technical Representative (COTR) who will be responsible for technical management of the procurement effort.

The procurement package should be submitted with sufficient time for completing the acquisition process. Lead times from submittal of the procurement package to contract award vary from 6 months to 1 year or more for competitive, negotiated contracts. Fixed-price contracts may be awarded within 2-3 months. The lead time is also influenced by the source evaluation methods used. For many competitive contracts requiring discussions with the offerors, a Source Evaluation Committee (SEC) may be used. An SEB is required for procurements in excess of $25 million. SEB policies and procedures are defined in the NASA FAR supplement 1815.300. The SEB process is concluded by an SEB review and report, followed by announcement of the decision by the Source Selection Official.

Funds must be properly authorized before a procurement process can be initiated, and the formal contract award cannot be completed until funds are obligated by an allotment authority. This will usually occur at the same time that implementation funds are released at the beginning of Phase C.

4.5 PHASE C - DESIGN

4.5.1 Phase C Team Composition

The size of the Phase C team may be easily double that of the Phase B team. As system designers produce detail hardware drawings and software specifications for the engineering or prototype model, more involved participation by fabrication, testing, and assembly personnel will be required. Represented functions are:

* Science/Technology
* Project management
* Systems engineering
* Subsystem management
* Experts in the major hardware fields related to the concept options surviving Phase A
* Systems analysts
* Software engineering
* Product assurance engineering
* Testing
* Project controls (costing, scheduling, and configuration management)
* Quality control
* Electronics manufacturing
* Hardware manufacturing
* Integration engineering
* Data management
* Design
* Operations

During Phase C and Phase D, the project team will reach its greatest size. Planning in the areas of data management and operations will begin as will the development of draft procedures for test and integration. Detail design activities are a large portion of the Phase C effort.

4.5.2 Phase C - Final Design and Engineering Development Implementation

This phase will produce final drawings for fabrication of the engineering model hardware and software. There may also be development and fabrication of flight hardware for long lead items.

The beginning of Phase C again requires a period of project planning. New project members must be briefed and contractors must be given a "period of understanding" time to staff their efforts and become familiar with the project. An important aspect of this is preparation of the contractor systems engineering management plan, which will define how the contractor's systems analysis will be conducted.

In-house systems engineering activities include a review and flow down of requirements in the form of design specifications for hardware configuration items (HWCI) and computer software configuration items (CSCI). Error budget refinements are ongoing and continual reallocation to subsystems will likely be necessary. Systems engineering also supports the preliminary design and analysis of subsystems and monitors the development of hardware and software for any long lead items. Continuing tradeoffs among cost, schedule, and expected system performance will be required. Plans for systems integration are developed and the end-to-end performance model is maintained and updated. Overall systems performance is monitored as more information becomes available through analysis and test.

Plans for manufacturing, testing, integration, and verification/validation are finalized. Systems engineering must closely monitor both hardware and software development and prototype testing, and upgrade the systems simulation model as subsystem design and performance characteristics become better defined. Software simulation models and mission simulation models should be completed during this phase.

More sophisticated breadboard and brassboard models of key subsystems and assemblies may be developed during this time. It is important that these models incorporate both hardware and software functions, although the actual hardware elements will not be of flight quality and the software implementation may use an informal pseudo-code rather than a higher order language. These models can demonstrate subsystem performance and verify the interfaces and performance assumptions of the systems simulation model.

An important task, which is continued in Phase C, is the performance of operational scenarios and mission anomaly simulations. These "what-if" analyses are completed by subjecting the overall systems simulation model to events and combinations of events which are off-nominal. The purpose is to develop contingency plans which will be available during mission operations, should the events actually occur. Note that these "what-if" analyses are performed early in the implementation phase to serve as an input to the system design.

As in earlier phases, the effect of a subsystem's inability to achieve required performance must be carefully examined. If a requirement at a low level cannot be met, the impact of this shortfall on overall system performance must be determined. The goals and requirements development process outlined in this handbook makes this analysis much easier. However, determination of appropriate corrective action will be the responsibility of the systems engineer. Suitable workarounds may be found through the trade study portion of the systems engineering process. Specifically, performance measures should be established for the trade study, options should be developed, and the performance of those options estimated on the performance measures. Subsystem level trade studies will form a major part of the Phase C systems engineering analysis effort.

The engineering, or prototype, model is built in Phase C from preliminary drawings. This model is built to the specifications expected for the final flight version, but without all the formal configuration and quality control. This model is produced to represent the best information on the final flight configuration anticipated. Software development efforts progress concurrently, leading to an initial version of the flight software. The engineering model provides an opportunity to identify and work around unforeseen problems with system fabrication and assembly. The model also allows an early indication of the expected performance of the final flight system. Shortcomings that are identified may be corrected and verified, with updates being input to the final Build-To Drawings. Configuration and change control will typically be the responsibility of the subsystem managers at this point. After CDR, software will be placed under configuration control with any changes reviewed internally by the software development organization. Any software changes, which are made due to a deficiency identified by team members not involved with soft-ware development, are reviewed by the project configuration control function.

The CDR should be scheduled when the detailed design and analysis of hard-ware and software is approximately 95 percent complete and when engineering model testing is complete. Following the CDR, any action items or changes are incorporated. Systems engineering must verify the effect of any changes on systems performance.

4.6 PHASE D - FABRICATION, INTEGRATION, TEST, AND EVALUATION

4.6.1 Phase D Team Composition

The functions represented in the Phase D team will closely resemble that of Phase C. The main difference is a shift in the focus of efforts from design into system production. Fabrication and procurement will assume a strong role as will configuration management and product assurance. Testing and integration will likewise come to the forefront as analysis activities decrease. The net result will be an effort that is roughly equivalent in magnitude to that of Phase C.

4.6.2 Phase D Implementation

This phase concentrates on proto-flight or flight hardware fabrication and on flight software implementation. Continual emphasis must be placed on verification of requirements at all levels. As parts are fabricated and assembled into subassemblies, assemblies, and subsystems, the integration plan must be implemented to impose appropriate test and verification requirements at each level. Error allocations should be verified and any Nonconformance-Failure Reports (NFR's) issued by quality assurance should be closely monitored to assess the effect on system tolerance and alignment budgets.

As subsystems are assembled into system elements and segments, increased emphasis is placed on the verification of external interfaces and the specification of test requirements to validate the system performance. A key aspect of the integration process is the integration of the computer software units at the appropriate stage of hardware assembly. Test procedures and system integration and operational procedures are prepared for verification testing at both subsystem and system levels.

Test Readiness Reviews (TRR's) are held to verify test readiness prior to critical tests. Systems acceptance is verified by a System Acceptance Review (SAR) prior to delivery of the system for integration at the next higher assembly level, or before integration to the launch vehicle.

4.7 PHASE E - OPERATIONS PHASE

4.7.1 Phase E - Pre-flight Team Composition

During Phase E, project emphasis shifts to preparation of the verified system for shipment and integration to the next higher level of assembly. Final data reduction and mission operations procedures are produced. By Phase E, the size of the project has begun to decrease, and many technical personnel may be relieved for other assignments. At this stage, the team may include:

* Science/Technology
* Project management
* Systems engineering
* Subsystem management
* Software engineering
* Product assurance engineering
* Project controls (costing, scheduling, and configuration management)
* Quality control
* Integration engineering
* Data management
* Operations

All activities required to allow for operation of the system in-orbit must be finalized and accomplished.

4.7.2 Phase E - Flight Mission Operations Team Composition

In the operational phase, control of the system is typically assumed by agents of the science or technology team representative. In some cases, LaRC personnel will be required to operate the systems from a ground station. In any event, the data gathered must be routed to the science/technology customer and processed into useful information. The project team is generally disbanded at this point with the exception of personnel to complete these activities.

4.7.3 Phase E - Preflight and Flight Mission Operations Implementation

The key systems engineering function during this phase is to monitor system performance during spacecraft/vehicle integration and pre-launch preparations. Additional systems engineering support may be needed to prepare for launch and flight operations. A key activity at this time is training of the flight operations team. All internal and external system interfaces are demonstrated during this phase using the Systems Integration Document (SID).

Control gates during this phase include the Operational Readiness Review (ORR) and the Flight Readiness Review (FRR). Systems engineering activities include the verification of the integrated system with the expected system performance as base-lined on the systems performance model.

Mission activities during this phase are focused on the initiation of flight operations and the validation of flight data. The first phase of this activity involves system performance checkout in-flight by the project team and initiation of routine flight operations. This is followed by routine flight operations by the flight operations team for as long as required, with only necessary involvement of the project team to address operational questions or flight anomalies. The crucial systems engineering activity is the consolidation of system records and the development of lessons learned information to support the systems engineering database. The concluding control gates for the project are the Operational Acceptance Review (OAR), or post-flight review which essentially summarizes the project activities and assesses mission success and the Lessons Learned Review (LRR). The purpose of the LRR is to collect and disseminate information on experiences gained during the project life time.

Chapter 5

SYSTEMS HARDWARE/SOFTWARE ENGINEERING

SUMMARY

This Chapter discusses the critical hardware/software systems interface, with emphasis on continuous communication and periodic formal briefings and reviews.

5.1 INTRODUCTION

The systems engineer must establish and maintain control of all systems interfaces throughout the project life cycle. This is especially true with the important hardware/software interface. Project schedule and flowcharts may show an initial definition of "system requirements" followed by an early division into "software requirements" and "hardware requirements." The software and hardware development processes must not be allowed to proceed on two separate and parallel paths until some point far downstream where the "hardware/software integration" occurs. Systems engineering should pay special attention to hardware/software questions and see that communication is initiated early and maintained between the system users, algorithm developers, hardware engineers, software engineers, and software development teams. It is important that the hardware/software interface be continually verified for consistency and completeness throughout the project life cycle.

An important aspect of the overall system design is to avoid premature allocation of systems requirements into hardware and software requirements. Particularly in the early phases of a study, there is usually a need to prepare project cost estimates. Cost analysts desire to learn systems details such as the weights of hardware and the number of lines of software code. Such estimates should be based on comparative data base experience from other projects when possible rather than from a premature "freezing" of the system design. It is very important that systems tradeoffs and optimization be allowed to proceed until systems scope and overall feasibility are demonstrated before partitioning the system into hardware and software modules.

The systems engineer should be aware of the full scope of the project software influence including applications for ground support equipment (GSE), engineering models and simulations, data processing, embedded architecture, test data reduction, and mission operations, as well as prototype and flight hardware. Unless other factors are present, considerations of language commonality, maintainability, and reusability should be emphasized to assure the most cost-effective selections for the overall system.

5.2 SYSTEMS REQUIREMENTS DEFINITION

The emphasis during definition of systems requirements should be on both hardware and software aspects of the system. For example, a systems design with a hardware bias might over-emphasize parameters such as instrument transfer functions or signal throughputs with less emphasis on important systems control and data flow characteristics.

It is important that the Systems Requirements Document include functional requirements, as well as physical requirements. One aspect of this is to address user operational needs very early in the formulation phases. This will establish how the user intends to operate the system, what type of commands and responses are expected, what type of message flows are needed, and what timing allocations are necessary. A very useful technique at this stage is the use of rapid prototyping to develop an end-to-end model of the system. When the end-to-end message flows are demonstrated, the system can evolve into a more detailed requirements model with state transition diagrams and specific control and data flows. This leads to the system architecture model which will subsequently be decomposed into specific hardware and software requirements.

5.3 SYSTEMS REQUIREMENTS ALLOCATION

It is desirable that the allocation of requirements be the result of studied deliberation among the systems, hardware, and software engineers. When agreement is reached on the division of systems responsibilities, it is the responsibility of the systems engineer to document the results in the Systems Requirements Document. These requirements are further flowed down into more detailed design specifications and assigned to hardware and software configuration items.

The layout of the systems concept must assure that each systems entity represents a logical and contiguous portion of the system. Each Computer Software Unit (CSU) should be a stand-alone entity that can be verified and validated. At each system level, it is important that systems requirements maintain traceability to hardware and to software code. Also, in the event of changes, it is important that the changes be flowed down from the systems level rather than be inserted from the bottom up.

5.4 HARDWARE/SOFTWARE DEVELOPMENT PROCESS

Systems engineering should maintain control and cognizance of the system while the separate hardware and software development processes are proceeding. At the top level this is done through periodic status reports and formal design reviews. Schedules should identify the critical paths between hardware and software and show key verification points. The systems engineer must also stay intimately connected to the design process at all times. One method of doing this is by maintaining the system simulation model and upgrading it to incorporate all design refinements. The systems engineer should also closely monitor the software verification and validation program which should be in continuous operation. By reviewing the periodic software metrics which are generated, a gauge of development progress is obtained. Hardware activities should also be closely monitored with emphasis on any nonconformance or failure reports and results of analytical verification, breadboard, and brassboard tests.

An important aspect of the hardware/software development process is continual change control. Beginning in Phase C both hardware and software designs will be under configuration management and should be functionally correlated at all times. An example of this is the development of engineering models and prototypes. The performance capability of a prototype will initially be limited, perhaps to a few critical functions, and then become more sophisticated as the model is developed. The software version developed to support the respective hardware model must provide an appropriate functional capability to demonstrate the system performance. As the model evolves and develops more functionality, the compatibility between hardware and software must be maintained until the model eventually becomes a fully functional simulation of the flight system.

Close surveillance of all contract end item (CEI) hardware should also be maintained to assure that technical requirements will be met. The systems engineer should carefully maintain the system error allocation budget and make adjustments as the design is refined.

5.5 HARDWARE/SOFTWARE INTEGRATION

If the hardware and software development processes have been adequately controlled, there should be few surprises during systems integration. As the system was decomposed during the development process, so it is recombined for integration from the part level through subassemblies to assemblies, and so forth in an iterative process. The respective CSU's are incorporated and verified at each level before being integrated into the major elements or segments of the system. This assures requirements traceability and isolation of any hardware/software incompatibilities or anomalies.

Systems performance is verified against the systems model baseline during each phase of the integration process and further verified by performance and environmental tests. The culmination of the process is the end-to-end systems test with subsequent review of test results and systems validation and acceptance.

Chapter 6

SYSTEMS ENGINEERING ACTIVITIES AND PRODUCTS

SUMMARY

This Chapter provides a detailed listing of project activities and products which form the systems engineering process as applied to each project life cycle phase.

6.1 INTRODUCTION

This Chapter includes a detailed list of activities and products. Emphasis is placed on the cyclical nature of the systems engineering process leading to progressive and more detailed refinement of the system requirements.

6.2 FORMULATION PHASE: PRE-PHASE A - PRELIMINARY REQUIREMENTS AND CONCEPTS ANALYSIS

PRE-PHASE A - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

1. PROJECT INITIALIZATION

Activities	Products
1. Request for Pre-Phase A study by sponsoring management	1. Request for support of required organizations
2. Establish Pre-Phase A Team	2. Assigned study team
3. Initial briefing to study team by sponsoring organization	3. Informal presentation
4. Develop project strategy	4. Pre-Phase A Work Plan
5. *Establish Project Data Base*	5. *Project Data Base*

PRE-PHASE A - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

2. USER NEEDS AND GOALS ANALYSIS

Activities	Products
6. Formal presentation to study team by sponsoring group 6.1 Document known project constraints and control items	6. Presentation materials
7. *Document science and technology goals*	7. *Statement of Project Status* *7.1 Project goals* *7.2 Experiment maturity* *7.3 Known constraints*
8. *Perform goals analysis* *8.1 Establish the relationship between science and technology goals and NASA / LaRC goals* *8.2 Establish impact and vision of the experiment*	8. *Preliminary Goals Analysis Document* *8.1 Project Justification Statement* *8.2 Project Impact / Vision Statement*
9. *Establish goals hierarchy*	9. *Goals Hierarchy*

3. SYSTEMS REQUIREMENTS AND CONSTRAINTS

Activities	Products
10. Document project requirements and constraints (concept independent) *10.1 Document science and technology requirements* *10.2 Develop instrument requirements* *10.3 Develop spacecraft requirements* *10.4 Develop mission, flight operations, and ground support operations requirements* *10.5 Develop preliminary hardware / software requirements* *10.6 Document management requirements* *10.7 Develop preliminary requirements verification and validation plan*	*10. Preliminary Systems Requirements Document* *10.1 Science and technical requirements* *10.2 Instrument requirements* *10.3 Spacecraft requirements* *10.4 Mission, flight operations, and ground operations requirements* *10.5 Hardware / software requirements* *10.6 Project requirements* *10.7 Requirement originator* *10.8 Requirement responsibility* *10.9 Initial requirements verification and validation plan; preliminary test plan*
11. Establish requirements database	*11. Requirements Data Base*

PRE-PHASE A - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

4. PERFORMANCE MEASURES

Activities	Products
12. *Document initial performance criteria measures from goals hierarchy* *12.1 Identify the performance criteria measures* *12.2 Determine the relative importance of the performance measures*	12. *Initial Performance Measures Statement* *12.1 Performance measures list* *12.2 Performance measures relative importance*

5. SYSTEMS CONCEPTS

Activities	Products
13. Develop preliminary alternate system concepts 13.1 Instruments 13.2 Spacecraft 13.3 Launch vehicle 13.4 Mission 13.5 Ground operations 13.6 Flight operations 13.7 Interfaces	13. Broad-based list of potential system concepts to satisfy experiment requirements
14. *Evaluate and document alternate system concepts*	14. *Preliminary Alternate Concepts List*

6. CONCEPTS ANALYSIS

Activities	Products
15. Characterize alternate system concepts *15.1 WBS* *15.2 Schedule* *15.3 Cost* *15.4 Risk* *15.5 Technical Performance Measures* *(concept modeling, simulation, functional analysis, engineering analysis techniques, as required)*	*15. Initial Alternate Concepts Analysis Document* *15.1 Preliminary WBS* *15.1 Preliminary schedule* *15.1 Preliminary cost estimate* *15.1 Preliminary risk assessment* *15.1 Preliminary interface definition, operations scenario, flight classification, performance measure estimates* *15.1 Required technology developments*

PRE-PHASE A - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

7. CONCEPTS RANKING

Activities	Products
16. Rank concepts based on performance measures *16.1 Perform trade studies* *16.2 Perform multi-attribute decision analysis*	*16. Concept Ranking Report* *16.1 Trade study results* *16.2 Decision analysis results* *16.3 Phase A Go / No Go recommendation*

8. SYSTEMS DEVELOPMENT

Activities	Products
17. Prepare Phase A Study Plan	*17. Proposed Phase A Study Plan* *17.1 Systems engineering management plan*
18. Prepare for Sponsoring Group Director's Review (SGDR)	18. SGDR Review Package

9. REVIEW, VERIFICATION, AND VALIDATION

Activities	Products
19. Sponsoring Group Director's Review	19. SGDR summary and recommendation

10. DECISION POINT

Activities	Products
20. Phase A Project Approval	20. Go/No Go for Phase A

6.3 FORMULATION PHASE: PHASE A - REQUIREMENTS DEFINITION AND CONCEPTUAL TRADE STUDIES

PHASE A - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

1. PROJECT INITIALIZATION

Activities	Products
1. Appoint Project Manager for Phase A Study	1. LaRC Announcement
2. Establish Phase A Team	2. Memorandum of Understanding between Division Chiefs
3. Review Phase A Study Plan	3. Final Phase A Study Plan
4. Review Pre-Phase A product	4. Phase A Team Building
5. Assign team responsibilities	5. Project Manager Memorandum
6. Develop business, management and technical plans 6.1 Prepare initial System Acquisition Plan	6. Phase A Work Plan and System Acquisition Plan

2. USER NEEDS AND GOALS ANALYSIS

Activities	Products
7. Review and update the goals analysis	*7. Review and update the goals analysis*

3. SYSTEMS REQUIREMENTS AND CONSTRAINTS

Activities	Products
8. Complete the project requirements and constraints *8.1 Develop the System Requirements Document* *8.2 Refine/define requirements verification and validation plan*	*8. Project requirements documentation* *8.1 Systems Requirements Document with Science, Instrument, Mission, Spacecraft, Hardware/Software, Operations and Project requirements* *8.2 Preliminary Verification and Validation Plan*

PHASE A - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

4. PERFORMANCE MEASURES

Activities	Products
9. Refine performance measures *9.1 Refine relative importance* *9.2 Develop decision analysis for each performance measure*	*9. Performance Measures Statement*

5. SYSTEMS CONCEPTS

Activities	Products
10. Review and refine alternate systems concepts	*10. Alternate Concepts List*

6. CONCEPTS ANALYSIS

Activities	Products
11. Perform a refined trade-off analysis on the alternate concepts *11.1 WBS* *11.2 Schedule* *11.3 Cost* *11.4 Risk* *11.5 Cost/Risk Analysis* *11.6 Concept technical simulation - (each technical performance measure will be evaluated through simulation models; the completeness of the model should reflect the parameters ranking)* *11.7 Develop probability distribution for each performance measure for each alternative concept as applicable*	*11. Alternate Concepts Analysis Document (All concepts)* *11.1 WBS* *11.2 Schedule* *11.3 Preliminary life cycle cost estimate* *11.4 Risk assessments* *11.5 Interface definitions* *11.6 Required technology development* *11.7 Results of simulation analysis for each technical performance measure* *11.8 Probability distribution for projected concept values on each performance measure*

7. CONCEPTS RANKING

Activities	Products
12. Perform multi-attribute decision analysis *12.1 Perform decision analysis* *12.2 Perform sensitivity analysis* *12.3 Decide feasibility of project*	*12. Decision Analysis Report* *12.1 Decision analysis results* *12.2 Sensitivity analysis results* *12.3 Phase B Go/No Go recommendation*

PHASE A - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

8. SYSTEMS DEVELOPMENT

Activities	Products
13. New Start Proposal *13.1 Prepare proposed Project Plan* *13.2 Perform risk assessment*	*13. Preliminary New Start Proposal* *13.1 Draft Project Plan with statement of project goals, justification, and impact* *13.2 Statement of ranked concepts* *13.3 Results of decision analysis* *13.4 Preliminary Risk Reduction Plan* *13.5 Preliminary Configuration Control Plan* *13.6 Mission Needs Statement (if req'd)*
14. Prepare Project Initiation Agreement (PIA)	14. Proposed PIA
15. Prepare Phase B Study Plan	15. Proposed Phase B Study Plan 15.1 Technical and Management Plan 15.2 Systems Engineering Management Plan
16. Prepare for reviews	16. Phase A Review Packages 16.1 PSRR Review Package 16.2 SEIRC Review Package 16.3 Group Directors' Review Package

9. REVIEW, VERIFICATION, AND VALIDATION

Activities	Products
17. Preliminary Systems Requirements Review	17. PSRR Committee summary and concurrence
18. Space-flight Experiment Initiatives Review Committee (SEIRC) Review	18. SEIRC summary and concurrence
19. Group Directors' Review	19. Management approval

20. LaRC Center Director's Review	20. Management approval to present to NASA Headquarters
21. NASA Headquarters Review of proposed project	21. NASA Headquarters approval to proceed to Phase B

PHASE A - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

10. DECISION POINT

Activities	Products
22. Phase B Project Approval	22. Project Initiation Agreement (PIA) approved

6.4 FORMULATION PHASE: PHASE B(1) - CONCEPT DEFINITION AND PRELIMINARY DESIGN

PHASE B(1) - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

1. PROJECT INITIALIZATION

Activities	Products
1. Obtain authorization for Project Phase B based on proposed Project Plan and PIA	1. Funding authorization from NASA Headquarters to LaRC
2. Appoint Phase B Project Manager	2. LaRC Announcement
3. Appoint Phase B Definition Team 3.1 Obtain commitments from Division Chiefs	3. Memorandum of Understanding between Division Chiefs
4. Update and finalize the proposed Phase B Study Plan 4.1 Technical and management *4.2 Systems Engineering Management Plan*	4. Phase B Study Plan 4.1 Technical and Management Plan *4.2 Systems Engineering Management Plan*

2. USER NEEDS AND GOALS ANALYSIS

Activities	Products
5. Review and finalize goals analysis	*5. Final Goals Analysis Document*

PHASE B(1) - PROJECT ACTIVITIES AND PRODUCTS

(Systems Engineering activities are shown in bold italics.)

3. SYSTEMS REQUIREMENTS AND CONSTRAINTS

Activities	Products
6. Review and update Systems Requirements Document *6.1 Place change control on Systems Requirements*	*6. Final Systems Requirements Document*
7. Review Project Justification, Vision, and Impact	*7. Final Goals Analysis Document*
8. Review systems requirements and constraints (Instrument, Spacecraft, Mission) *8.1 Mission, orbit, launch, operations, recovery scenarios defined* *8.2 Electrical* *8.3 Thermal control* *8.4 Structural* *8.5 Attitude control* *8.6 Power system* *8.7 Command and data handling* *8.8 Communications* *8.9 Hardware and Software*	*8. Final Systems Requirements Document*

4. PERFORMANCE MEASURES

Activities	Products
9. Review and update performance measures and relative importance	*9. Final Performance Measures Statement*

5. SYSTEMS CONCEPTS

Activities	Products
10. Review and update alternate concepts	*10. Final Alternate Concepts List*

PHASE B(1) - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

6. CONCEPTS ANALYSIS

Activities	Products
11. Review and update Alternate Concepts Analysis Report *11.1 Risk assessment survey and technology development*	*11. Final Alternate Concepts Analysis Report*

7. CONCEPTS RANKING

Activities	Products
12. Update decision analysis	*12. Final Decision Analysis Report*
13. Select baseline systems concept	*13. Baseline Systems Concept Selected*

8. SYSTEMS DEVELOPMENT

Activities	Products
14. Expand and refine baseline design; perform subsystem tradeoff analyses *14.1 Identify major segments, elements, sub-systems* *14.2 Allocate requirements to segments, elements, and subsystems* *14.3 Refine system design* *14.4 Initiate risk reduction activities*	*14. Baseline Systems Concept Package* *14.1 Preliminary systems decomposition and definition* *14.2 Requirements allocation* *14.3 Final Systems Requirements Document* *14.4 Risk Reduction Plan, Failure Modes and Effects Analysis (FMEA), and Critical Items List (CIL)*

14.5 Expand WBS to subsystem level *14.6 Expand life cycle cost analysis* *14.7 Expand schedule* *14.8 Expand interface definition* *14.9 Expand verification and validation plan* *14.10 Prepare preliminary Interface Control Document* *14.11 Allocate requirements to mission operations, instrument, and control software* *14.12 Initiate Product Assurance Plan* *14.13 Prepare final Configuration Control Plan* *14.14 Prepare Preliminary Parts List*	*14.5 Refined WBS* *14.6 Updated life cycle cost estimate* *14.7 Updated schedule* *14.8 Final Systems Requirements Document* *14.9 Updated Verification and Validation Plan* *14.10 Preliminary Interface Control Document (ICD)* *14.11 Software Requirements Spec.* *14.12 Prelim. Product Assurance Plan* *14.13 Initial Configuration Ctrl. Plan* *14.14 Preliminary Parts List*
15. Produce preliminary design drawings	15. Preliminary Design Drawings

PHASE B(1) - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

8. SYSTEMS DEVELOPMENT (Continued)

Activities	Products
16. Develop systems integration document	*16. Systems Integration Document* *16.1 Systems Integration Plan* *16.2 Spacecraft Integration Plan* *16.3 Launch Vehicle Integration Plan* *16.4 Spacecraft Site Test Plan*
17. Prepare Review Packages	17. Review Packages for Systems Requirements Review (SRR), Non-Advocate Review (NAR), Software Concept Review (SCR), Software Requirements Review (SRR), Preliminary Design Review (PDR), Software Preliminary Design Review
18. Expand/update proposed Project Plan	18. Proposed Project Plan

9. REVIEW, VERIFICATION, AND VALIDATION

Activities	Products
19. Systems Requirements Review (SRR) and Software Concept Review (SCR)	19. SRR and SCR Committee summary and recommendations reports
20. LaRC Review of NAR Package for Non-Advocate Review Committee	20. Management approval
21. NAR Review for Non-Advocate Committee at NASA Headquarters	21. Non-Advocate Committee summary and recommendations report
22. Conduct Software Requirements Review (SRR)	22. SRR Committee summary and recommendations report
23. Conduct Preliminary Design Review (PDR) and Software Preliminary (Architectural) Design Review (SPDR) for a 70 percent complete design	23. PDR & SPDR Committee summary and recommendations report

10. DECISION POINT

Activities	Products
24. Obtain all management commitments for new start support and resources	24. Approval for new start in Phase C

6.5 FORMULATION PHASE: PHASE B(2) - SOURCE SELECTION PROCESS

PHASE B(2) - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities are shown in bold italics.)

Activities	Products
1. Complete Systems Acquisition Plan 1.1 Identify items for procurement 1.2 Identify items for in-house build 1.3 Identify Government Furnished Property (GFP)	1. System Acquisition Plan
2. Establish procedures and identify areas to receive, calibrate, and store Contract End Items (CEI's) and GFP	2. Product Assurance Plan
3. Prepare preliminary Statement of Work (SOW) or Technical Specifications for review by potential bidders	3. Preliminary SOW or Technical Specifications

4. Prepare Procurement Package for submittal to Acquisition Division	4. Approved Procurement Package
5. Confirm funds appropriated, budgeted, programmed	5. Approved Purchase Request
6. Develop and implement Source Selection Plan	6. Source Selection Plan
7. Evaluate Proposals	7. Source Evaluation Board (SEB) or Source Evaluation Committee (SEC) report and recommendation
8. Select Contractor	8. Source Selection Official (SSO) selection
9. Negotiate/Award Contract	9. Contract Award
10. Contractor Kickoff Meeting and implement "period of understanding" with Contractor	*10. Contractor Kickoff Meeting*

6.6 IMPLEMENTATION PHASE: PHASE C - FINAL DESIGN AND ENGINEERING DEVELOPMENT

PHASE C - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
1. Update and maintain the Systems Engineering Data Base	*1 . Systems Engineering Data Base*
2. Review and augment Phase C project team; appoint mission operations team	2. Phase C Project Team; Mission Operations Team
3. Review and update Phase B products	3. Brief new project members
4. Update the WBS	4. Complete WBS
5. Update Preliminary Parts List	5. Preliminary Parts List
6. Update System Acquisition Plan--buy vs. build components	6. System Acquisition Plan
7. Allocate subsystem error budgets	*7. Error Allocation Plan*
8. Develop system interface specification documents	*8. Interface Control Document (ICD) update*
9. Develop design specifications for Contract End Items (CEI's)	9. Design specifications for CEI's
10. Start procurement of CEI's--hardware and software	10. Approved Purchase Requests
11. Start acquiring Government Furnished Property	11. Approved Purchase Requests
12. Finalize Configuration Control Plan	12. Final Configuration Control Plan

13. Develop design specifications for in-house build items	*13. Design specifications for in-house build items; Specifications Traceability Matrix*
14. Support risk reduction activities; develop plans for engineering models	*14. Risk Reduction Plan; Engineering Model Build and Procurement Plan*
15. Update verification and validation plan	*15. Verification and Validation Plan; Performance Verification Matrix*
16. Iterate design and analysis to produce detail system design	16. Design Analysis Reports
17. Produce "Build-To" Drawings"	17. Build-To" drawings

PHASE C - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
18. Start fabrication of engineering and prototype models (hardware and software)	18. Working Models
19. Finalize plans 19.1 Manufacturing (fabrication and assembly) 19.2 Testing *19.3 Verification and validation* *19.4 Integration* 19.5 Operations 19.6 Support systems 19.7 Facilities 19.8 Software production 19.9 Calibration 19.10 Science 19.11 Product assurance	19. Final plans 19.1 Manufacturing (fabrication and assembly) 19.2 Testing *19.3 Performance Verification Matrix* *19.4 Systems Integration Document* 19.5 Mission Operations 19.6 Ground Support Systems 19.7 Facilities 19.8 Software Production 19.9 Calibration 19.10 Science 19.11 Product Assurance
20. Refine and baseline 20.1 Life cycle cost analysis 20.2 WBS 20.3 PERT	20. Phase C Project Plan 20.1 Life Cycle Cost Analysis 20.2 WBS 20.3 PERT
21. Test engineering and prototype hardware and software models	21. Engineering Model Test Reports (continues through Phase D)
22. Complete systems simulation model for verification of model tests and detailed analysis	*22. End-to-end Systems Simulation Model* *22.1 Instrument Software* *22.2 Ground Support Software* *22.3 Operations Command Software* *22.4 Modeling and Analysis Software*
23. Perform mission anomaly simulation	*23. Operational Contingency Plan*
24. Schedule CDR and SCDR for hardware/ software design 95 percent	24. CDR Panel appointed

complete	
25. Prepare CDR package	25. CDR Package
26. Conduct Critical Design Review (CDR)	26. CDR Committee Report and Recommendations
27. Conduct Software Critical Design Review (SCDR)	27. SCDR Committee Report and Recommendations
28. Incorporate CDR RFA's into design and write CDR report	28. CDR Report
29. Update "Build-To" drawings	29. "Build-To" Drawings

PHASE C - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
30. Put Configuration Control into effect	30. Configuration Control Plan
31. Begin fabrication of long lead, in-house build, hardware and software flight items	31. Fabrication Control Plan
32. Management approval to begin Phase D	32.Memorandum from Chairperson, CDR and SCDR Committees

6.7 IMPLEMENTATION PHASE: PHASE D - FABRICATION, INTEGRATION, TEST, AND EVALUATION

PHASE D - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
1. Complete external CEI acquisition activities; update Parts List	1. Updated Parts List; approved Purchase Requests
2. Assure readiness of facilities for assembly/ test of subsystems and system (clean rooms)	2. Facilities Plan
3. Monitor and update PERT chart with emphasis on acquisition, transportation, and integration of subsystems	3. Assembly/Integration Procedures
4. Establish contingency plans for high-risk components	*4. Operational Contingency Plan*
5. Fabricate in-house build hardware and implement software units	5. Update Fabrication Plan

6. Perform subsystem verification and conduct design reviews as needed	6. Ad-hoc subsystem design reviews; update hardware and software design
7. Review/refine all testing documents (parts, subsystems, and systems)	7. Final Test Plan
8. Review / refine the system integration documents	*8. Update Systems Integration Document, Systems Test Procedures, Site Spacecraft Test Plan*
9. Monitor and coordinate risk reduction activities	*9. Update Risk Reduction Plan*

PHASE D - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
10. Update all operations documents (ground and flight)	10. Update Mission Operations Plan and Procedures
11. Assure readiness of support equipment	11. Update Ground Support Equipment Operations Plan and Procedures
12. Continuously update interface changes	*12. Update ICD*
13. Receive CEI's; inspect, test, log, and locate in bonded stores; reorder as required	13. Product Assurance Plan; End Item Logbooks
14. Prepare part and subsystem test procedures as required	14. Test Procedures
15. Conduct system and subsystem test coordination meetings (TCM) and test readiness reviews (TRR's) as required	15. Test Readiness Reviews
16. Assemble parts into subassemblies and assemblies and integrate software	16. Assembly Instructions and Drawings
17. Test subassemblies and assemblies	17. Test Reports
18. Evaluate test results and verify systems performance per allocated requirements and error budgets	*18. Update Performance Verification Matrix*
19. Feedback test results *19.1 Introduce design changes and redesign as required* *19.2 Adjust interfaces as required* *19.3 Refabricate* *19.4 Retest* *19.5 Update PERT*	*19. Verified Parts, Subassemblies, and Assemblies*
20. Assemble subassemblies and assemblies into subsystems and integrate software	20. Assembly Instructions and Drawings
21. Test subsystems	21. Subsystem Test Report

22. Evaluate test results and verify systems performance per allocated requirements and error budgets	*22. Update Performance Verification Matrix*

PHASE D - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
23. Feedback test results 23.1 Introduce design changes and redesign as required 23.2 Adjust interfaces as required 23.3 Re-fabricate 23.4 Retest 23.5 Update PERT	23. Verified Subsystems
24.Update and finalize systems test procedures	*24. Systems Test Procedures*
25. Conduct system Test Coordination Meeting (TCM), Test Readiness Review (TRR), and Software Test Readiness Review as required	25. Test Readiness Reviews
26. Assemble subsystems into the next level elements, segments, or system	26. Assembly Procedures and Drawings
27. Test system 27.1 Qualification test 27.2 Systems performance test 27.3 Ground truth verification test	27. System Test Report
28. Evaluate test results and verify systems performance per allocated requirements and error budgets	*28. Update Performance Verification Matrix*
29. Feedback test results *29.1 Redesign as required* *29.2 Adjust interfaces as required* *29.3 Refabricate* *29.4 Retest* *29.5 Update PERT*	*29. Verified and validated functioning system (or Functional Configuration Audit [FCA])*
30. Show compliance to Verification and Validation Plan; update verification matrix	*30. System Verification Report*
31. Document "as-built" configuration	31. As-built Documentation (or Physical Configuration Audit [PCA])
32. Create the systems maintenance manuals	*32. Systems Maintenance Document*
33. Update the mission operations manuals (SAR)	33. Update Mission Operations Plan

PHASE D - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
34. Conduct System Acceptance Review (SAR) and Software Acceptance Review	34. Systems Acceptance Report
35. Document all system and GSE transportation requirements and procedures	35. Transportation and Receiving Procedures
36. Ship System to integration site	36. Ship system

6.8 PHASE E - OPERATIONAL PHASE: PREFLIGHT AND FLIGHT MISSION OPERATIONS AND DISPOSAL

PHASE E - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
1. Mission Operations Team begins training a Project Operations Control Center (POCC) and Data Processing Center (DPC)	1. Mission Operations Team Training
2. Deliver all applicable plans and procedures to the Operations Project Manager	2. Final Mission Operations Plan and Procedures
3.Train Project Operations Control Center (POCC), Mission Control Room (MCR), and Data Processing Center (DPC) personnel for operations duties	3. Final Mission Timelines and Operational Scenarios
4. Update Systems Integration Document and Spacecraft Site Test Plan	*4. Systems Integration Document, Spacecraft Site Test Plan*
5. Finalize all system receiving, buildup, checkout, testing, launch preparation, and mission timelines/procedures	5. Final Mission Operations Procedures
6. Move all key personnel to the MCR, POCC, and DPC sites	6. Flight Operations Team Ready

7. Inspect and update MCR, POCC, and DPC facilities	7. Completed Operations Facilities

PHASE E - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
8. Transport all spacecraft handling and checkout equipment to the launch site	8. Ground Support Equipment Operations Plan
9. Transport system and spacecraft to launch site	9. Pre-Launch Preparations Plan
10. Conduct Operational Readiness Review (ORR)	10. ORR Committee Report and Recommendation
11. Receive system 11.1 Inspect 11.2 Buildup 11.3 Checkout 11.4 Life test	11. Operating System at Launch Site
12. Integrate/assemble system and spacecraft 12.1 Checkout 12.2 Test 12.3 Calibrate 12.4 Perform data flow check to POCC and DPC	12. Operating Spacecraft at Launch Site
13. Integrate/assemble spacecraft and launch vehicle 13.1 Checkout 13.2 Verify interfaces 13.3 Test	13. Assembled Launch Vehicle
14. Perform data flow checks and proper communications to MCR, POCC, and DPC	14. Verified Data Flow
15. Perform system maintenance as required	15. Pre-flight Operational Procedures
16. Conduct Flight Readiness Review (FRR)	16. Flight Readiness Review Report
17. Launch spacecraft 17.1 Countdown 17.2 Launch 17.3 Orbit insertion	17. Launch Procedures

PHASE E - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
18. Certify mission/system operational readiness 18.1 Spacecraft/system hardware/software operating properly 18.2 Ground hardware/software operating properly 18.3 Data communications functioning 18.4 Development Project Manager's review complete 18.5 POCC Team in place 18.6 Exceptions to any of the above evaluated and documented	18. Mission Operational Readiness Certificate
19. Transfer system control from MCR to POCC	19. Mission Operational Procedures
20. Begin mission operations	20. Functional System in Orbit
21. Establish communications between the spacecraft, POCC, and DPC	21. Spacecraft Operational
22. Determine the status of the mission and spacecraft; evaluate the effects of anomalies on the experiment: 22.1 Operations 22.2 Data Products 22.3 Schedules, and so forth	22. Mission Verification
23. Establish workaround plans for all known anomalies	23. Mission Recovery Plans and Procedures
24. Perform the experiment operations start-up procedure	24. Experiment Operational
25. Carry out the experiment real-time life and health test	25. Experiment Verification
26. Perform scheduled flight operations as planned	26. Flight Operations Procedures
27. Conduct Operational Acceptance Review (Post-flight Review)	27. Post-flight Review Report
28. Start mission science and engineering continuous data trending	28. Data Operations Procedures
29. Resolve experiment anomalies; establish and execute contingency operating plans as needed	29. Experiment Recovery Plans and Procedures
30. Manage allocation of spacecraft resources	30. Flight Operations

PHASE E - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
31. Generate scheduled and special mission reports	31. Mission Reports
32. Generate the experiment science and technology data 32.1 Raw data 32.2 Calibrated data 32.3 Engineering data 32.4 Science data	32. Data Reports
33. Validate data	*33. Data Validation Report*
34. Monitor the engineering performance of the spacecraft and the experiment	34. Flight Operations
35. Perform system maintenance as needed	35. Flight Operations
36. Accomplish planned data distribution	36. Data Operations Procedures
37. Conduct ongoing mission operations performance evaluations	37. Flight Operations
38. Conduct missions final engineering test and generate test report	38. Engineering Test Report
39. Terminate mission 39.1 Perform final data processing 39.2 Terminate flight operations according to plans 39.3 Shut down the communications network 39.4 Shut down ground operations elements	39. Mission Termination
40. Recover spacecraft and experiment if applicable	40. Recovery Operation
41. Deliver recovered data (tapes, memory, disk, and so forth) to DPC	41. Recovery Operation
42. Deliver flight hardware to cognizant authorities for disposition/storage	42. Recovery Operation

PHASE E - PROJECT ACTIVITIES AND PRODUCTS
(Systems Engineering activities shown in bold italics.)

Activities	Products
43. Create Post Mission Evaluation Report 43.1 Show validation of requirements 43.2 State overall system performance 43.3 Evaluate the effects of anomalies 43.4 Discuss lessons learned	*43. Mission Evaluation Report*
44. Conduct Lessons Learned Review (LLR)	44. Lessons Learned Report
45. Continue science and engineering data reduction	45. Data Operations Plan
46. Implement Pre-Planned Product Improvement (P3I) if applicable	46. Refurbish and Modify Equipment
47. Consider Next Operational Concept (NOC)	47. Follow-on Proposal; Restart Systems Engineering Process at Requirements Definition and Conceptual Trade Studies - Phase A

Chapter 7

LESSONS LEARNED

SUMMARY

This Chapter contains information and experience gained through application of systems engineering at Langley Research Center.

7.1 INTRODUCTION

Information in this Chapter was compiled from experience with application of the systems engineering process at Langley Research Center (LaRC). Observations are listed point by point and this Chapter will be expanded as more information becomes available. Note that the systems engineering database will contain lessons learned information.

7.2 COMMUNICATION

The importance of good communications cannot be overstressed. The systems engineer must verify that mechanisms are in place to assure that all team members are in close contact with one another. The most critical aspect of this task is the communication of the customer's needs and goals to the rest of the team. Likewise, frequent briefings to management of the sponsoring organization are extremely helpful in keeping the project on track. A thorough understanding must be maintained by the subsystem managers of not only their own requirements, but also of the relationships existent with other sub-systems. Regular team meetings are strongly indicated to facilitate good communication. In addition, the systems engineer must pay careful attention to those related team members whom should communicate frequently because of closely correlated subsystems, and ensure that regular interchanges occur. This aspect of the process must be exercised for local as well as off site team members.

7.3 FORMULATION STUDY STAFFING

In Pre-Phase A and Phase A, resources are usually provided by the sponsoring organization. Typically, the resources will be very limited, if they exist at all. Likewise, study team members will be participating in the study in addition to their regular duties. Often, a meeting of all team members will be difficult to schedule without a long lead-time. This makes the job of the systems engineer especially challenging if the study is to be performed in a timely manner. A proactive Study Manager is helpful, but ultimately, the systems engineer must gather information for systems analyses from all participating team members. There are three approaches that the systems engineer may take to address this situation.

First, it is a very good idea to establish a routine team meeting date at the very beginning of the study, such as "every Wednesday at 3 p.m." Next, tools should be used to maximize the efficiency of the information captured during these meetings. Tape or video recorders, electronic copy boards, and cameras can be invaluable in ascertaining what was said or meant in a meeting. This allows the systems engineer to review meetings for detail, without having to re-contact participants. This is also very helpful in recording decisions for the audit trail. Finally, the systems engineer should frequently visit the other team members as necessary to retrieve information needed for analysis. Taken together, these three steps can significantly decrease the time required for formulation studies.

7.4 AUDIT TRAIL

Maintenance of good project records is advisable for several reasons. First of all, a good audit trail can be invaluable when quick justification for decisions is required. This can be especially valuable when contractor disputes or grievances arise. Also, complete and well organized records can prevent duplication of effort by the project team. Finally, the information serves as a project archive that may be utilized as a reference for future endeavors and for generation of lessons learned. The audit trail is essentially the information stored as the project data base and includes such items as: conceptual layouts, schematics, meeting minutes, analysis summaries and results, correspondence, schedules, and the typical information stored on the system configuration. Establishment and maintenance of the project audit trail is a prime responsibility of systems engineering, since much of the critical information is generated by that function.

7.5 RESOURCE BUDGET RESERVES

The nature of research projects is one of uncertainty, since most undertakings have never before been accomplished. Thus, risk is inherent in LaRC project development, and planning for the unexpected is necessary. One method of planning for the unforeseen is to hold a significant portion of system resources in reserve in the early stages of the project. After the baseline concept is established, system commodities such as weight and power must be allocated to the control of subsystem managers by the systems engineer. While no exact numbers are available, experience has shown that budget reserves of 30 percent or so are well advised in the Preliminary Design Review (PDR) time frame of a project. This value should be varied according to the risk associated with a given development. For example, a well characterized, second generation system may require only a 10 percent budget reserve at PDR. Conversely, a project breaking new technological ground may need 40 percent at this stage. Some reserve should be held even at the system Critical Design Review (CDR) to deal with uncertainties through the fabrication and assembly process. Reserves in the 10 percent range are typically sufficient for an average LaRC project at CDR. Even with these margins, the Project Manager and the systems engineer must be frugal in the release of reserves, to prevent abuse of the process. Subsystem projections should be studied carefully to assure that resource requirements are realistic, and not underestimated because of an understanding that additional resources are available for the asking.

Appendix A

LIST OF ACRONYMS

SUMMARY

This Appendix provides a list of acronyms used in this handbook. Detailed definitions may be found in Appendix B, "Glossary."

AMSD	Aerospace Mechanical Systems Division
AO	Announcement of Opportunity
APA	Allowance for Program Adjustment
ATP	Authority to Proceed
CAE	Computer-Aided Engineering
CDR	Critical Design Review
CEI	Contract End Item
CIL	Critical Item List
CoDR	Conceptual Design Review
COTR	Contracting Officer's Technical Representative
CSC	Computer Software Component
CSCI	Computer Software Configuration Item
CSU	Computer Software Unit
DPC	Data Processing Center
EHB	Engineering Handbook
EMI	Electromagnetic Interference
FCA	Functional Configuration Audit
FFT	Fast Fourier Transform
FID	Field Installation Director
FIR	Finite Impulse Response
FMEA	Failure Mode and Effects Analysis
FRR	Flight Readiness Review
GDR	Group Directors' Review
GFP	Government Furnished Property
GSE	Ground Support Equipment
HFI	Host Field Installation
ICD	Interface Control Document
IIR	Infinite Impulse Response
IRD	Interface Requirements Document
JSC	Johnson Space Center
KSC	Kennedy Space Center
LaRC	Langley Research Center
LASE	Lidar Atmospheric Sensing Experiment
LIDAR	Light Detection And Ranging
LLR	Lessons Learned Review
LAPD	Langley Policy Directive
LAPG	Langley Procedures and Guidelines
LRR	Launch Readiness Review
MCR	Mission Control Room
MNS	Mission Needs Statement
NAR	Non-Advocate Review
NFR	Nonconformance Failure Report
NPD	NASA Policy Directive
NPG	NASA Procedures and Guidelines
NOC	Next Operational Concept
NSTS	National Space Transportation System

NSY	New Start Year
OAR	Operational Acceptance Review
OBS	Organizational Breakdown Structure
OMB	Office of Management and Budget
ORR	Operational Readiness Review
OSEMA	Office of Safety, Environment and Mission Assurance
PAA	Program Associate Administrator
PCA	Physical Configuration Audit
PDR	Preliminary Design Review
PIA	Project Initiation Agreement
POCC	Payload Operations Control Center
POP	Program Operating Plan
PSRR	Preliminary Systems Requirements Review; Pre-Shipment Readiness Review
RFA	Request For Action
RTOP	Research and Technology Operating Plan
SAR	Software Acceptance Review; System Acceptance Review
SCDR	Software Critical Design Review
SCR	Software Concept Review
SEAL	Software Engineering and Ada Laboratory
SEB	Source Evaluation Board
SEC	Source Evaluation Committee
SEIRC	Space-flight Experiment Initiatives Review Committee
SEMP	Systems Engineering Management Plan
SGDR	Sponsoring Group Director's Review
SID	Systems Integration Document
SPDR	Software Preliminary Design Review
SPF	Single Point Failure
SQL	Structured Query Language
SRR	Software Requirements Review; Systems Requirements Review
SSO	Source Selection Official
STRR	Software Test Readiness Review
TBD	To Be Determined
TCM	Test Coordination Meeting
TQM	Total Quality Management
TRR	Test Readiness Review
VLSIC	Very Large-Scale Integrated Circuits
WBS	Work Breakdown Structure

Appendix B

GLOSSARY OF TERMINOLOGY

SUMMARY

This Appendix contains an alphabetical listing of systems engineering and project terminology used in this handbook with working definitions and related acronyms.

Administrative Requirements - Those non-technical conditions, such as cost and schedule constraints, which are imposed on a project.

Advocate - A person who speaks and writes in support of the science or technology goals of a program.

Allowance for Program Adjustment (APA) - Resources allocated for: expansions in project requirements resulting from NASA Headquarters approved changes in project objectives or scope; the resolution of unforeseen major problems; project stretch outs from Agency funding shortfalls, etc. These resources are managed by the NASA Headquarters Program Office.

Alternate Concepts Analysis List - Summary of tradeoff studies and quantitative comparison of the alternate systems concepts initially prepared in Pre-Phase A and finalized in Phase B.

Alternate Concepts List - Summary of potential systems concepts generated during Pre-Phase A as the basis for further design tradeoff studies.

Alternative Concept - One of a number of generalized system configurations that is analyzed against the technical, budgetary, and schedule requirements and goals of a project and summarized in the Concept Analysis Document.

Announcement of Opportunity - The process by which proposed investigations are solicited for a future space flight.

As-Built Drawings - An updated version of the Build-To Drawings which reflect any alterations incurred in fabrication and are completely consistent with the hardware as it exists.

As-Built Project Baseline - The in-house configuration baseline which constitutes the actual flight hardware, flight software, and ground support end items as delivered.

As-Designed Project Baseline - The in-house preliminary configuration baseline which defines the hardware and software during the implementation phases. The preliminary baseline document is subject to continuous change control and is updated formally at reviews as the system configuration is incrementally developed.

Assembly - A number of parts, or subassemblies, joined together to form a complete unit or article which can perform a specific function.

Baseline - A set of documents that define an item and are formally designated and fixed at a specific time in the project life cycle, serve as the basis for further development, and are changed only by formal change control procedures.

Baseline Systems Concept - The baseline system design selected at the end of Phase B and proposed for implementation in Phase C.

Brassboard - A hardware assembly of preliminary circuits or parts to prove out a specific function. More sophisticated than breadboards, brass-boards begin to approach the challenge of form and fit, as well as function.
Breadboard - A hardware assembly of preliminary circuits or parts used to prove the feasibility of a device, circuit, system, or principle. It is a function-only model with no attempt to have form or fit.

Budget - The available amount of an operational system commodity such as weight, power, volume, heating capacity, or cooling capacity.

Build-To Drawings - Engineering drawings used to fabricate a given part, assembly, or subsystem.

Computer Software Component (CSC) - A functional or logically distinct part of a computer software configuration item.

Computer Software Configuration Item (CSCI) - A collection of software elements treated as a unit for the purpose of configuration management.

Computer Software Unit (CSU) - The smallest logical entity specified in the design of a computer software component and the actual entity in code that implements a testable aspect of the requirements.

Concept Analysis Document - A document reporting the results of the alternate concept studies, typically including WBS, schedule, cost, and technical performance measures estimates.

Concept Ranking Report - A document reporting the order in which the alternate concepts best fulfill the project requirements, including justification for the ranking.

Concepts List - See Alternate Concepts List.

Conceptual Design Review (CoDR) - A formal critique of the system occurring in Phase B to determine the adequacy of the overall system conceptual design and project progress. See Systems Requirements Review.

Configuration Item (CI) - A collection of parts treated as a unit for configuration control. Also referred to as Computer Software Configuration Item (CSCI) and Hardware Configuration Item (HWCI).

Configuration Management - The process of identifying and defining the deliverable product set in a system, controlling the release and change of those items throughout the life cycle, recording and reporting the status of product items and change requests, and verifying the completeness and correctness of the product items.

Configured Hardware/Software - An arrangement of various elements of computer hardware and software integrated in such a manner as to best satisfy particular needs.

Constituency - Those persons or organizations that will be served by the project and affected by fulfillment of the top goal.

Constraints - The restrictions and boundaries that place limitations on the system development, such as budget, schedule, performance, and so forth.
Contingency Plans - Recovery procedures to be followed to minimize damage in the event that a major risk concern proves valid.

Contract End Item (CEI) - An article that is purchased under project acquisition control.

Contracting Officer's Technical Representative (COTR) - Primary technical adviser to the Contracting Officer and the person responsible for technical direction of the Contractor.

Control Gate - A formal review conducted to evaluate status and to approve that the project may proceed according to the project plan.

Cost - The consumption of resources expressed in terms of dollars or time.

Critical Design Review (CDR) - A formal critique of the system occurring in late Phase C to determine the adequacy of the overall system design and project progress, prior to complete fabrication and integration of the flight hardware. A series of reviews may be held to address specific segments or lesser entities of the system hierarchy.

Critical Items List (CIL) - An analysis derived from the Failure Modes and Effects Analysis which identifies the rationale or justification for retaining critical items.

Data - Unprocessed facts, figures, and measurements gathered during the course of a project that includes the goals, activities, and results.

Data Validation Report - Post-flight analysis and report to validate the operational science and technology data in accordance with the requirements of the Verification and Validation Plan.

Decision Analysis - A structured procedure for determining the best of a number of feasible alternatives relative to project objectives.

Decision Analysis Report - Summary report issued in Phase A to rank potential systems concepts and assess the feasibility of proceeding to Phase B. This is reviewed and updated in Phase B.

Decision Point - See Control Gate.

Derived Requirements - Additional requirements identified as additional system architecture/design is introduced in the system design process. Also referred to as specified requirements.

Design Specifications - Documentation delineating a precise, detailed, and verifiable description of the pertinent design parameters needed to purchase or fabricate a given hardware part, software unit, or configuration item.

Development Cost Commitment (DCC) - The cost ceiling established by the Administrator for the total costs to be incurred in Phase B through Phase D of the project life cycle.

Element - A complete, integrated set of subsystems capable of accomplishing an operational role or function.

Engineering Model - A complete very high fidelity form, fit, and functions hardware model which resembles the flight article in all respects but which does not necessarily have flight quality parts. It is not generally used as the final qualification article.

Error Allocation Plan - Document which allocates the system level error budget to segments, elements, and subsystems.

Experiment Maturity - The extent of development of the project experiment indicating the level of confidence that the science goals can be successfully accomplished.

Failure - The inability of a system or part to perform in accordance with specification requirements; a functional test or operating discrepancy.

Failure Modes and Effects Analysis (FMEA) - A procedure by which each potential failure mode in a system or subsystem is identified and the results or effects on the system are determined.

Flight Classification - The formal category assigned to a project that delineates its relative importance to national space goals or its relative hazard severity. See Payload Classifications.

Flight Hardware - All physical/material equipment that will be launched to satisfy project objectives.

Flight Operations - See Mission Operations.

Flight Readiness Review (FRR) - Formal review to assess the overall readiness of the project to perform its science/mission objectives.

Flight Software - All computer programs, including computer system operations and mission dedicated applications used for control, communications, data acquisition, analysis, and so forth.

Formulation Phases - The initial stages of the project life cycle when the emphasis is on requirements analysis, project planning, concept definition, feasibility demonstration, and preliminary design. Includes Pre-Phase A, Phase A, and Phase B.

Functional Analysis - A structured approach to the description of a system whereby the operation is progressively broken down to its lowest tasks for the purpose of system analysis and synthesis.

Functional Configuration Audit (FCA) - The control gate that verifies the acceptance test results are consistent with the test requirements previously approved at the PDR and CDR.

Functional Requirements - Those system requirements which express what system capabilities are necessary in order to achieve a goal.
Goal - A qualitative statement of the project's basic purpose and desired end result.

Goals Analysis - A structured examination of the statement of project goals, initially occurring at project inception.

Goals Analysis Document - A report that gives details of the project goals. The contents include the project background, vision, project goals hierarchy, and identified constraints. May include the Performance Measures Statement.

Goals Hierarchy - A diagrammatic representation of project goals starting with the most general purpose, progressing to more specific goals, and terminating with project requirements.

Ground Operations - All components of the project related to the conduct of integrated logistics support including the interfaces with the users, sustaining engineering, preflight/ post-flight data processing, and transportation services.

Group Directors' Review - A formal presentation made to the LaRC Group Directors for the purpose of obtaining endorsement to proceed to the LaRC Center Director's Review, and ultimately to NASA Headquarters.

Host Field Installation - The NASA field installation participating in a multiple field installation program which is responsible for providing the institutional resources for the Program Manager function.

Impact - The anticipated scientific, technological, and social changes that would occur due to the successful completion of the project or the change in performance, schedule, or cost due to the modification of requirements.

Implementation Phases - The final stages of the project life cycle when hardware and software items are designed, fabricated, integrated, verified; undergoes preflight operations; becomes operational; and the mission is completed. Includes Phase C, Phase D, and Phase E.

Information - Data which has been processed in such a way as to provide an increased understanding of an area of interest.

In-House Project - A project conducted on-site or in the immediate vicinity of a field installation in which essentially most major technical, business, and management tasks normally performed by a prime contractor are performed by the installation's civil service staff.

Instrument - The portion of the flight system--hardware and software--responsible for the measurement and capture of the data needed to satisfy the science and technology requirements.

Instrument Requirements - Those requirements imposed on the instrument for the purpose of achieving the science/technology goals by obtaining the necessary measurements and data.

Interface Control Document - The document which defines the internal and external interface design to a level of detail sufficient to integrate and assemble the system.

Interfaces - The common boundaries and connections between various portions of a system.

LaRC Center Director's Review - A formal presentation made to the LaRC Center Director for the purpose of obtaining authority to pursue funding from NASA Headquarters.

Launch Readiness Review (LRR) - Review held to assess readiness of the launch vehicle, spacecraft, and ground systems for launch and space flight. Also referred to as Preflight Review. See Flight Readiness Review.

Launch Vehicle - The physical means, such as various rocket systems or the National Space Transportation System (NSTS), whereby the flight spacecraft is placed into space.

Lessons Learned Review (LLR) - A final review to collect and disseminate information on experiences gained during the project lifetime and to provide an overview of the lessons learned.

Measurements - Physical characteristics recorded during a mission for the purpose of providing information relevant to the science goals.

Memorandum of Understanding (MOU) - A formal documented agreement between two parties which prescribes the specific roles and responsibilities of cooperative efforts. Also referred to as Memorandum of Agreement (MOA).

Mission - The specific sequence of events which the flight system must execute in order to accomplish the goals of the project.

Mission Need Statement (MNS) - The document that establishes the justification for undertaking an Agency objective or effectively pursuing an opportunity pertaining to an Agency objective. It is the document that grants authority to initiate a Phase B effort for a candidate project.

Mission Operations - Those activities necessary to carry out the mission flight plan from pre-launch through landing or termination.

Mission Operations Plan - A composite set of planned space and ground support operations.

Mission Requirements - Those requirements imposed on the flight system that describe what the system must accomplish in order to achieve the science/technology goals.

Modeling - The evaluation of a system concept by creating a mathematical representation of the system characteristics and simulating the conditions of interest.

NASA Headquarters Review - A formal presentation made to NASA Headquarters personnel for the purpose of obtaining funding for a particular project.

NASA Project Life Cycle - All phases of a NASA project including research, development, design, evaluation, production, test, deployment, operation, data analysis, validation, and disposal.

New Start Proposal - Proposal prepared at the end of Phase B for presentation to NASA Headquarters for the purpose of obtaining Phase C project funding.

New Start Year (NSY) - Year designated in the NASA budget cycle for an approved project to receive funding to begin the implementation phases of the effort.

Non-Advocate Package - Materials and information presented to the NASA Headquarters Associate Administrator's non-advocate committee for the purpose of independent assessment of the feasibility and worth of the proposed project. A review of a proposed major system project by a non-advocate team appointed by and reporting to the Deputy Administrator. The Non-Advocate Review team is comprised of experienced project management, technical, and budget personnel drawn on an ad hoc basis from organizations that will not participate in the implementation of the proposed project. These reviews provide Agency management with independent assessments of the adequacy of the project formulation effort.

Nonconformance - A nonfunctional workmanship condition of any item (hardware or software) in which one or more characteristics does not conform to drawing specification or procedure requirements; a fabrication or assembly discrepancy reported on NASA Langley Form 143, "Nonconformance-Failure Report (NFR)."

Objectives - Specific, quantitative, and verifiable tasks that lead to accomplishment of the overall project goal.

Operational Acceptance Review (OAR) - Review held after the spacecraft is in orbit to verify systems performance prior to acceptance of the system for operational use. Also known as the Post-flight Review.

Operational Contingency Plan - A risk reduction tool to establish contingency plans for high-risk components.

Operational Readiness Review (ORR) - Review held at the program level to assure readiness for operation of the integrated spacecraft for preflight testing and operations.

Part - The smallest individual piece of a subassembly which cannot be disassembled. The selection of parts is dictated by the payload classification and is subject to the NASA Standard Parts Program.

Payload Classifications - The NASA designation classifying each program/payload according to the criticality of the mission to NASA and national objectives, program cost, and the acceptable level of risk of a partial or complete failure. The payload classifications are:

Class A - High Priority, Minimum Risk
Class B - High Priority, Medium Risk
Class C - Medium Priority, Medium/Low Risk
Class D - High Risk, Minimum Cost

Performance Measure Relative Importance - Weighing factors which indicate which performance measures are of greater or lesser criticality to the customer. These factors are used in decision analysis.

Performance Measures - Variable parameters which are used to determine the degree to which alternative concepts can satisfy the project goals. The attributes used for configuration trade studies.

Performance Measures Statement - A listing and explanation of the performance measures including their relative importance. May be contained in the Goals Analysis Document. Also referred to as Performance Measures Document.

Performance Requirements - A requirement that specifies a performance characteristic that a system must possess to achieve a project goal.

Performance Verification Matrix - See Verification and Validation Plan.

Phase A - Preliminary Analysis (Requirements definition and conceptual trade studies) - The second segment in the NASA/LaRC project life cycle in which requirements are refined and candidate designs are investigated in depth. Phase A culminates with a preferred system concept and plans for development of that concept, including a Mission Need Statement when required.

Phase B(1) - Definition (Concept definition and preliminary design) - The stage of the project life cycle where the baseline system concept is established with understanding of the full range and implications of implementing the proposed project sufficient to make an Agency commitment. The PDR is accomplished in Phase B.

Phase B(2) (Source selection process) - The stage of the project life cycle designated for the acquisition of major contracted support or contract end items.

Phase C - Design (Final design and engineering development) - The stage of the project life cycle where the system is completely analyzed and designed, test and verification plans are defined, and hardware and software models are developed. The CDR is accomplished in Phase C.

Phase D - Development (Fabrication, integration, test, and evaluation) - The stage of the project life cycle where the system hardware and software elements or parts are purchased, developed, or fabricated; assembled; tested; and the integrated system is verified, validated, and delivered.

Phase E - Operations (Preflight and flight mission operations and disposal) - The project life cycle stage in which the developed system is prepared for flight, placed in orbit, and operated until the completion of its mission and then shut down or disposed.

Phase Project Team - Those individuals that have been assigned by LaRC management to participate in a given segment of the project.

Phase Study Plan - The delineation of a detailed strategy and schedule for accomplishing established objectives of a given life cycle phase.

Physical Configuration Audit (PCA) - A control gate that verifies the physical configuration of the product that corresponds to the "build-to" (or "code-to") documentation previously approved at the CDR.

Physical Requirements - A requirement that specifies a physical characteristic that a system must possess.

Preliminary Design Drawings - The initial design drawings that are presented at the Preliminary Design Review.

Preliminary Design Review (PDR) - A critique of the system occurring in Phase B to determine the adequacy of the overall system preliminary design and project progress. A series of reviews may be held to address specific segments or lesser entities in the system hierarchy.

Preliminary Systems Requirements Review (PSRR) - Review held early in Phase A to demonstrate that preliminary systems requirements have been defined.

Pre-Phase A (Preliminary Requirements and Concepts Analysis) - The initial stage of the NASA/LaRC project life cycle in which required system performance, project constraints, and candidate system concepts are defined, analyzed, and documented.

Pre-Shipment Readiness Review (PSRR) - A review held after the completion of the major acceptance testing and prior to shipment to assure the readiness of the flight system and project/mission plans. See System Acceptance Review.

Principal Investigator (PI) - The primary customer for research developments.

Product Assurance Plan - A document, developed by the project product assurance engineer, which implements a product assurance program including safety, reliability, and quality control.

Program - A related series of long term efforts directed toward a broad scientific or technical goal and funded by NASA Headquarters appropriations.

Program Associate Administrator (PAA) - The NASA Headquarters official responsible and accountable for formulation and implementation of a major system program.

Program Commitment Agreement (PCA) - The contract between the Administrator and the cognizant PAA for implementation of a major system program.

Program Cost Commitment (PCC) - The cost ceiling established by the Administrator for the life cycle costs of a major system program.

Program Evaluation and Review Technique (PERT) - A formal, structured method for monitoring and controlling the scheduling and integration of a project.

Program Operating Plan (POP) - Time phased projection of Center resource utilization; includes narrative describing planned activities and over-guideline fiscal requirements.

Programmatic Performance Measure - A parameter, such as life cycle cost or development time, which is used as an indication of the extent to which non-technical project goals have been satisfied.

Project - Normally an element of a program, a project is an activity with clearly defined team membership, objectives, schedule, and cost intended to gain knowledge, create a capability, or provide a service. A project includes the definition, design, development, fabrication, verification, operations, data analysis and distribution, information extraction, technology transfer, and disposal of a system.

Project Data Base - A structured collection of information that captures and relates the detailed history, goals, requirements, resources, results, system configuration, and so forth, of the project.

Project Goals - The statement of the project's basic purpose and desired end result together with the technical and administrative objectives targeted for achievement.

Project Impact/Vision Statement - A written statement of the anticipated scientific, technological, and social changes that would occur due to the successful completion of the project.

Project Initiation Agreement (PIA) - An agreement between the Program Associate Administrator and the Director of a Center in charge of the project. The PIA outlines a new project's management and technical strategies, acquisition plan, schedule, resource estimates, cost, contingency reserves, and all other project ground rules. The Project Initiation Agreement is superseded by the approved Project Plan.

Project Justification Statement - A written statement giving reasons as to why pursuit of the science goals is a worthwhile endeavor.

Project Manager - The field installation official who is exclusively responsible for project definition and implementation to completion within a given set of boundary conditions (technical, cost, schedule, and organization approach).

Project Plan - The document prepared by the field installation that establishes the overall plan for implementation of the project. The Project Plan emphasizes the management and programmatic aspects of the project rather than technical information, and establishes the agreement(s) between the PAA and the involved FID's (Single Field Installation Programs), or between the program manager at the HFI and the field installation project managers (Multiple Field Installation Programs).

Project Requirements - The constraints and performance measures, derived from project goals, which must be satisfied by the developed system to achieve project success.

Project Requirements Review (PRR) - Review held during Phase B to demonstrate the completion of systems requirements definition and flow-down prior to the start of preliminary design. See Systems Requirements Review.

Proto-flight - A flight quality hardware article that is used for qualification but not tested to the point of destruction. Following any refurbishment that may be required, the proto-flight article is flown in space.

Prototype - A flight quality hardware test article used for final qualification. It is generally tested beyond expected life limits and is, therefore, not refurbishable for flight. Also refers to a software model constructed for derivation and demonstration of customer requirements and performance assessment and which may evolve into production software.

Requirement - A condition or capability needed to achieve an objective. See also: Derived Requirements, Functional Requirements, Performance Requirements, Physical Requirements, Project Requirements, Science Requirements, Technical Requirements.

Requirement Originator - The person or organization that has created or requested a stated requirement.

Requirement Responsibility - The accountability for assuring that a stated requirement is implemented and verified.

Requirements Data Base - A structured collection of information that captures, organizes, and relates the project goals, detail requirements, and system configuration.

Requirements Validation - The process of evaluating the project development process to ensure compliance with the stated requirements.

Requirements Validation Plan - The delineation of a detailed strategy and schedule for accomplishing the validation of the project requirements. See also Verification and Validation Plan.

Risk - The likelihood of an undesirable event occurring and the severity of the consequences of the occurrence. The product of the probability of an undesired event and the consequences (usually in dollars) should the event occur.

Risk Assessment - The process of determining the source, probability, and severity of events which are hazardous to the project.

Risk Reduction Plan - System level plan to assess, track, and reduce risk from the conceptual definition stage through systems verification. Also referred to as the Risk Management Plan.

Robust - Strongly formed or constructed. Having performance margin and relatively insensitive to variations in environmental parameters.

Schedule - A graphic or tabular portrayal of project activities, their duration, and their relationships to each other. See Program Evaluation and Review Technique.
Science Goal - The basic purposes and desired end result of the science experiment.

Science Plan - The document prepared by the project Principal Investigator, which describes how the results of system operation will be converted to useful scientific knowledge.

Science Requirements - The functionality required of a system in order to satisfy an identified science goal.

Segment - A major portion of a system comprised of a grouping of elements and computer software configuration items that are closely related and often physically interface.

SEIRC (Space-flight Experiment Initiatives Review Committee) Review - A review held at LaRC at the end of Phase A, under the chair of the Head, Space Projects Office, for the purpose of independently assessing the compatibility of a project with the LaRC mission, the readiness of a project to proceed, and the capability of LaRC to support the implementation of the project.

Sensitivity Analysis - A detailed review of the effects on system performance due to variations in environmental parameters. Also used in decision analysis to refer to a review of the impact of variations in performance measure relative importance on the overall ranking of various system options.

Simulation - A computerized mathematical model of a physical system that can be used to predict the performance and response of the system to a given stimulus under specific conditions.

Single Point Failure (SPF) - A portion of the system which, if failure of that portion occurs, total system failure will result.

Software Acceptance Review (SAR) - Review or series of reviews held at the end of Phase D to demonstrate the completion of software development and to formally accept the as-built software baseline.

Software Concept Review (SCR) - A review held early in Phase B and in conjunction with the Systems Requirements Review to evaluate the software conceptual design and operational concept for economic and technical feasibility.

Software Critical Design Review (SCDR) - Review held at the end of Phase C and in conjunction with the system CDR to review the software detailed design document, unit test procedures, the user's guide, and operational procedures manual and data base design to establish that the applicable data requirements have been satisfied. A series of reviews may be held to address individual system segments or lesser entities in the system hierarchy.

Software Preliminary Design Review (SPDR) - Review held at the end of Phase B and in conjunction with the PDR to establish that the hardware/software interfaces have been defined and that the software preliminary design has been adequately completed. A series of reviews may be held to address individual system segments or computer software configuration items.
Software Requirements Review (SRR) - Review held in Phase B subsequent to the Systems Requirements Review to demonstrate that software requirements are compatible with system requirements, to establish the adequacy of the software Verification and Validation Plan, and to assess the system architecture to determine adequacy, completeness, and achievability of the system requirements.

Software Requirements Specification - The document that defines the system software specification based on an analysis and risk assessment of preliminary systems requirements to ensure that software requirements are feasible, complete, and consistent.

Software Test Readiness Review (STRR) - Review to evaluate the readiness of software items to undergo performance and verification acceptance testing.

Software Verification and Validation Plan - Overall approach used to verify and validate software across the entire project life cycle.

Source Evaluation Board or Committee - Committee responsible for establishing the criteria to be used in evaluating proposals, administering those criteria, and recommending the Contractor selection.

Spacecraft - That part of the system that supports the science instrument by providing power, stability, thermal control, structural support, and communications.

Space flight Site Test Plan - Document describing the intended instrument ground test activities on board the spacecraft.

Space-flight Experiment Initiatives Review - See SEIRC (Space-flight Experiment Initiative Review Committee) Review.

Specifications - See Design Specifications.

Specifications Traceability Matrix - Matrix used to trace design specifications to their precedent systems requirements.

Sponsoring Group Director's Review (SGDR) - A formal presentation made at the end of Pre-Phase A, to officials from the Group in which a new experiment is being developed, for the purpose of presenting the Phase A study plan and obtaining approval to proceed into Phase A.

Sponsoring Organization - The LaRC unit from which a new project originates.

Sponsoring Organization Overview Presentation - A formal offering by the sponsoring organization to the systems engineering supporting function that describes in detail the goals and requirements of the system development, and the background and current status of related work.

Statement of Project Goals - The official documentation of the project's basic goals.

Statement of Project Status - A project document produced in the early part of Pre-Phase A that states the maturity of the science experiment and consequently the initial estimate of the situation and project status. May be contained in the Goals Analysis Document.

Statement of Work (SOW) - Itemization of tasks to be accomplished in order to satisfy project and performance requirements. Typically forms the basis for con tracts with external entities.

Subassembly - Two or more parts joined together to form a unit which is only a portion of a complete assembly.

Subsystem - A functional grouping of assemblies that combine to perform a major function within an element.

Support Equipment - All items required in the operation or testing of the development end items that are not an integral part of the flight system.

System - The combination of elements that must function together to produce the capability required to meet the mission need. The elements include all hardware, software, equipment, facilities, personnel, and the processes and procedures needed for this purpose.

System Acceptance Review (SAR) - Formal review to ensure that there is a high level of confidence that the flight item has complied with mission requirements and specifications, that it will be transported safely to its destination, and that it will operate as designed upon arrival.

System Acquisition Plan - The project plan which defines the approach for acquiring major system hardware and software segments and which addresses the "make or buy" question to specify which items will be purchased and which will be developed in-house.

System Requirements Document - See Systems Requirements Document.

System Verification Report - Summary report to show compliance of the "as-built" system with the requirements of the Verification and Validation Plan.

Systems Analysis and Design Procedure - An iterative systems engineering process used to define system goals, requirements, concepts, and development in a systematic and verifiable sequence.

Systems Engineering - A function that guides the transformation of customer/user needs into a flight system that meets the technical performance requirements within NASA/LaRC policy.

Systems Engineering Data Base - A project oriented data base used to control, allocate, and track systems requirements and documentation.

Systems Engineering Process - A set of iterative activities which applies systems engineering to a project.
Systems Integration Document - Formal plan for the integration of subsystems, elements, and segments into the system and subsequent integration of the system into the spacecraft and launch vehicle.

Systems Requirements Document - A written report clearly delineating the requirements of the project. A preliminary document is created in Pre-Phase A and the final Systems Requirements Document is produced early in Phase B.

Systems Requirements Review (SRR) - Review to define the project objectives and confirm that the system requirements are sufficient. A concept is presented which will identify subsystems and their resource allocations. Successful completion of the SRR baselines the science/mission objectives and subsystem allocations and approves the initiation of the preliminary design.

Systems Simulation Model - A theoretical or simulated mathematical model of the system that accepts inputs, produces outputs, and performs the specified functions of the system.

Technical Performance Measure - A parameter which is used as an indication of the extent to which the technical science/project goals have been satisfied. See also Performance Measures.

Technical Requirements - A condition or capability needed to satisfy a physical requirement, constraint, technical goal, or technology development.

Test Plan - A document prescribing the approach for intended test activities.

Test Readiness Review (TRR) - A review or series of reviews to evaluate the readiness of the system or parts of the system to undergo performance and environmental testing.

Trade Study - An analytical process that compares critical parameters of various alternate concepts for the purpose of determining the preferred configuration. Also referred to as tradeoff analysis.

Validation - The process that assures that the system will comply with the science and performance requirements.

Verification - The process to establish that subsystems and individual configuration items comply with specifications.

NOTE: Verification vs. validation. The significant difference between validation and verification should be made distinct. Verification consists of proof of compliance with specifications and can be demonstrated by test, analysis, inspection, or similarity. Conversely, validation consists of various proofs that the system accomplishes or can accomplish its goals. Validation is accomplished at the system level; verification is accomplished throughout the entire system architectural hierarchy.

Verification and Validation Plan - Systems level plan for system hardware and software verification and validation by test, analysis, inspection, or similarity. Includes the Performance Verification Matrix used to track system verification criteria. See also Software Verification and Validation Plan.

Verified Assemblies - Integrated parts and subassemblies that have been analyzed and tested in order to prove that the unit adheres to requirements.

Verified Subsystems - Integrated entities that have been analyzed and tested in order to prove that the subsystem adheres to requirements.

Verified System - Integrated subsystems that have been analyzed and tested in order to prove that the system adheres to requirements.

Vision - The anticipated scientific, technological, and social changes that would occur due to the successful completion of the project. Includes impact of successful project completion and consideration of technology transfer and utilization.

Work Breakdown Structure (WBS) - A hierarchical structured diagram that delineates the tasks to be performed to accomplish the project goals and the organizations that are responsible for those tasks.

Appendix C

ENGINEERING AND SYSTEMS ENGINEERING SOFTWARE

SUMMARY

This Appendix provides a brief description of typical commercial software commonly used in various engineering and systems engineering disciplines.

The sole intention of this Summary is to introduce to prospective users a few features of some of the available off-the-shelf computer software packages. It is not intended to introduce this list as inclusive nor up to date and users are encouraged to search the market for other packages that will best suit their particular needs. A description of a limited number of commercial packages is provided in alphabetical order for the following categories:

* Data Base
* Dynamic Analysis and Design tools
* Matrix Computation with graphics capability tools
* Symbolic Manipulation systems
* Systems Engineering support tools

ORACLE

Data Base

APPLICATION

ORACLE is a fully relational data base management system. The system is based on the Structured Query Language (SQL) environment that provides an extremely powerful capability for searching the data base. ORACLE allows the user to develop custom applications and can store a very large amount of information when compared to others. It can be used to develop applications to provide information to project management and manage configuration management.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS personal computers (PC's),
Sun 4/SPARC, variety of IBM and VAX platforms

ATTRIBUTES

ORACLE is menu driven and is constructed with application modules around the system kernel. These modules provide search and query capabilities, report generation tools, and graphic devices. ORACLE also has a "C" language interface that allows the user to connect the data base to programs developed off line. The system is highly portable allowing an application developed on one platform to be transported to another.

ADDITIONAL REQUIREMENTS

SOURCE

ORACLE Corporation
500 Oracle Parkway
Redwood Shores, CA 94065
(415)506-7000

TRACER

Data Base

APPLICATION

TRACER is a relational data base oriented tool supporting capture of requirements in a central data base, tracing of relationships among requirements and documents, tracing of change impact and approval status, production of status reports for managers and custodians, production of hard copy and electronic documents, and delivery of documents to electronic access via LAN. TRACER supports projects involving more than 25,000 requirements organized as a hierarchical set of documents, and is particularly useful in large projects.

SYSTEM AVAILABILITY

IBM AT or equivalents

ATTRIBUTES

TRACER is a database for requirements which provides instant electronic access. The tool provides a mechanism for change management, and is amenable to networking with a shared database. TRACER's printout identifies unsubstantiated requirements so they can be eliminated.

ADDITIONAL REQUIREMENTS

SOURCE

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, CA 91109

Also:

COSMIC
The University of Georgia
382 East Broad Street
Athens, GA 30602-4272
(706)542-3265

DADS

Dynamic Analysis and Design

APPLICATION

DADS (Dynamic Analysis and Design System) is a mechanical Computer-Aided Engineering (CAE) software package that enables the simulation and analysis of complex mechanisms and mechanical systems. Multidisciplinary applications such as controls, hydraulics, and mechanical systems are incorporated in DADS, which includes flexible and rigid body elements.

SYSTEM AVAILABILITY

PC's: 486 based MS-DOS personal computers. Workstations: SGI, DEC, SUN, HP, IBM. Mainframes: IBM, DEC. Supercomputers: CRAY, Convex

ATTRIBUTES

DADS solves for displacement, velocity, acceleration and reaction forces of models. It performs static, kinematic, dynamic, and inverse dynamic analysis. It allows users to model real world behavior and interprets performance through plots, graphs, tables, and animation. DADS animation is especially suited for demonstration and simulation of designs to verify concepts and performance. Other features of DADS include: feedback, controls, hydraulics, and flexible bodies animation.

ADDITIONAL REQUIREMENTS

SOURCE

CADSI
2651 Crosspark Road
Coralville, IA 52241
(319)626-6700

NASTRAN

Dynamic Analysis and Design

APPLICATION

NASTRAN (NASA Structural Analysis) is a general purpose program that analyzes most kinds of structures and constructions by using the displacement method of the finite element approach. It provides engineers with a range of modeling and analysis capabilities. NASTRAN offers structural and modeling elements that represent common types of structural building blocks such as rods, beams, shear panels, plates, and shells of revolution. More general types of building blocks can be represented by combining these simple elements or by using a general element capability. NASTRAN allows users to incorporate the effects of control systems, aerodynamic transfer functions, and other nonstructural features in the solution of the structural problem.

SYSTEM AVAILABILITY

386 and 486 based with Interactive UNIX operating system PC's, Sun 4/SPARC, HP workstation, DEC workstation, IBM mainframe, CDC Cyber, CRAY

ATTRIBUTES

The system handles several types of analysis, including static response to concentrated and distributed loads, thermal expansion and enforced deformations, dynamic response to transient and steady-state sinusoidal loads, and random excitation. The system determines real and complex eigenvalues for use in analyzing vibration and dynamic and elastic stability. NASTRAN has limited capability for solving nonlinear problems, including piecewise linear analysis of nonlinear static response and transient analysis of nonlinear dynamic response.

ADDITIONAL REQUIREMENTS

SOURCE

The MacNeal-Schwendler Corporation
815 Colorado Boulevard
Los Angeles, CA 90041-1777
(213)258-9111, (800)336-4858

MATLAB

Matrix Computation

APPLICATION

MATLAB is a high performance interactive software program for scientific and engineering numeric computation. It combines numerical analysis, matrix computation, signal processing, and graphics with a user interface in which problems and solutions are expressed in standard math notation.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Macintosh, Sun 4/SPARC, Apollo, VAX/VMS

ATTRIBUTES

MATLAB provides interactive access to state-of-the-art linear algebra and matrix algorithms from LINPACK and EISPACK as well as other numeric techniques. MATLAB functions include: differential equation solution, polynomial operations, matrix computation, and complex arithmetic as well as powerful signal processing tools, such as 1-D and 2-D FFT's, spectral analysis, and digital filtering. In addition to its comprehensive set of scientific functions and matrix operations, MATLAB is fully extensible, allowing users to edit existing functions and to create new ones. To view data graphically, MATLAB provides 2-D linear, log, semi-log, and polar plots, and 3-D mesh and contour graphs.

ADDITIONAL REQUIREMENTS

SOURCE

MathWorks, Inc.
Cochituate Place, 24 Prime Park Way
Natick, MA 01760
(508)653-1415

MATLAB -
CONTROL SYSTEM TOOLBOX

Matrix Computation

APPLICATION

MATLAB Control System Toolbox works with MATLAB to provide comprehensive functionality for control system design and analysis. It includes "classical" transfer function and "modern" state space control techniques, plus time domain and frequency domain responses, feedback gain selection, and model properties calculation.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS personal computers, Macintosh, Sun 4/SPARC, Apollo, VAX/VMS

ATTRIBUTES

The toolboxes are delivered as MATLAB M-files, so the user can see the algorithms and implementations, as well as make changes or create new functions to address a particular application.

ADDITIONAL REQUIREMENTS

MATLAB

SOURCE

MathWorks, Inc.
Cochituate Place, 24 Prime Park Way
Natick, MA 01760
(508)653-1415

MATLAB -
OPTIMIZATION TOOLBOX

Matrix Computation

APPLICATION

The MATLAB Optimization Toolbox contains a set of functions that implement the most widely used methods for performing minimization or maximization on general linear and nonlinear functions. For nonlinear minimization, this includes functions for unconstrained and constrained minimization, minimax, nonlinear least squares, and multi-objective and semi-infinite minimization. Functions for linear programming, quadratic programming, non-negative least squares, and solving nonlinear equations are included.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Macintosh, Sun 4/SPARC, Apollo, VAX/VMS

ATTRIBUTES

The Optimization Toolbox is implemented with MATLAB numeric computation software, which assures accurate results, fast performance, and portability. Most of the toolbox functions are delivered as MATLAB M-files, allowing the user to view the algorithms and implementations, as well as to make changes or create new functions to address a particular application.

ADDITIONAL REQUIREMENTS

MATLAB

SOURCE

MathWorks, Inc.
Cochituate Place, 24 Prime Park Way
Natick, MA 01760
(508)653-1415

MATLAB -
SIGNAL PROCESSING TOOLBOX

Matrix Computation

APPLICATION

The MATLAB Signal Processing Toolbox works with MATLAB numeric computation software for 1-D and 2-D digital signal processing and time series analysis. The toolbox includes 1-D and 2-D FFT's and inverses, FIR and IIR filter design, filter response and simulation, and power spectrum estimation.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Macintosh, Sun 4/SPARC, Apollo, VAX/VMS

ATTRIBUTES

The Signal Processing Toolbox is delivered as MATLAB M-files, allowing the user to view the algorithms and implementations, as well as to make changes or create new functions to address a particular application.

ADDITIONAL REQUIREMENTS

MATLAB

SOURCE

MathWorks, Inc.
Cochituate Place, 24 Prime Park Way
Natick, MA 01760
(508)653-1415

MATLAB -
SIMULINK

Matrix Computation

APPLICATION

SIMULINK is a powerful, interactive package for modeling, analyzing, and simulating dynamic nonlinear systems. SIMULINK employs a graphical, mouse driven interface, based on the X Window System.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Macintosh, Sun 4/SPARC, Apollo, VAX/VMS

ATTRIBUTES

SIMULINK supports linear, nonlinear, continuous time, discrete time, multivariable, multirate, and hybrid systems; models can be defined either as block diagram structures or by sets of differential equations. Fully integrated with the MATLAB numeric computation system, SIMULINK is suited for a broad range of system simulation problems.

ADDITIONAL REQUIREMENTS

MATLAB

SOURCE

MathWorks, Inc.
Cochituate Place, 24 Prime Park Way
Natick, MA 01760
(508)653-1415

MATLAB -
SPLINE TOOLBOX

Matrix Computation

APPLICATION

The MATLAB Spline Toolkit contains a set of functions for the construction and use of piecewise polynomial functions.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Macintosh, Sun 4/SPARC, Apollo, VAX/VMS

ATTRIBUTES

The Spline Toolbox is implemented with MATLAB numeric computation software, which assures accurate results, fast performance, and portability. Most of the toolbox functions are delivered as MATLAB M-files, allowing the user to view the algorithms and implementations, as well as to make changes or create new functions to address a particular application.

ADDITIONAL REQUIREMENTS

MATLAB

SOURCE

MathWorks, Inc.
Cochituate Place, 24 Prime Park Way
Natick, MA 01760
(508)653-1415

MATLAB -
SYSTEM IDENTIFICATION TOOLBOX

Matrix Computation

APPLICATION

The MATLAB System Identification Toolbox works with MATLAB numeric computation software package to build mathematical models of dynamic systems, based on observed I/O (input/output) data. Central features include parametric and non-parametric techniques for all phases of the system identification process: parametric estimation, spectral analysis, simulation, and presentation.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Macintosh, Sun 4/SPARC, Apollo, VAX/VMS

ATTRIBUTES

The System Identification Toolbox is delivered as MATLAB M-files, allowing the user to view the algorithms and implementations, as well as to make changes or create new functions to address a particular application.

ADDITIONAL REQUIREMENTS

MATLAB

SOURCE

MathWorks, Inc.
Cochituate Place, 24 Prime Park Way
Natick, MA 01760
(508)653-1415

MATRIXx

Matrix Computation

APPLICATION

MATRIXx is a comprehensive mathematical analysis tool for engineers and scientists. It is an interactive matrix manipulation environment that combines the powerful numerical tools of LINPACK and EISPACK with an easy to use interface, comprehensive graphics facility, and an expandable function library.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Sun 4/SPARC, IBM 739, IBM 930, HP UNIX workstations, VAX/VMS

ATTRIBUTES

MATRIXx is the entry-level tool for the ISI product family, which includes comprehensive systems analysis and control design, as well as nonlinear simulation, block diagram system modeling, and automatic real time code generation and implementation. MATRIXx features include:

* Fast matrix calculator.
* Comprehensive math programming environment.
* Fast computations.
* Extensive 2-D and 3-D graphics with Postscript output.
* Extensive and extensible mathematical function library.
* Intuitive, user friendly interface.

ADDITIONAL REQUIREMENTS

FORTRAN compiler

SOURCE

Integrated Systems, Inc.
3260 Jay Street
Santa Clara, CA 95054-3309
(408)980-1500, (800)675-MATH

MATRIXx -
CONTROL DESIGN MODULE

Matrix Computation

APPLICATION

The Control Design Module, used with MATRIXx, performs sophisticated classical and modern control design and analysis including SI/SO (single input/single output), MI/MO (multiple input/multiple output), and multivariable applications.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Sun 4/SPARC, IBM 739, IBM 930, HP UNIX workstations, VAX/VMS

ATTRIBUTES

The Control Design Module contains a comprehensive set of functions for controls and systems engineers, including frequency and time response calculations, model reduction, root locus plots, and Kalman Filter design. The Control Design Module has the following features:

* Classical tools
* Modern tools
* System representation
* Conversions
* System construction
* Time responses
* Steady state analysis

ADDITIONAL REQUIREMENTS

FORTRAN compiler, MATRIXx

SOURCE

Integrated Systems, Inc.
3260 Jay Street
Santa Clara, CA 95054-3309
(408)980-1500, (800)675-MATH

MACSYMA

Symbolic Manipulation

APPLICATION

MACSYMA is an interactive expert system and programming environment designed to assist in solving a wide spectrum of mathematical problems. MACSYMA offers symbolic and numeric manipulation and solution capabilities in algebra, calculus and numerical analysis, 2-D and 3-D report quality graphics, interfaces with mathematical text processors, and a user programming environment.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Sun 4/SPARC, IBM, HP UNIX workstations, Silicon Graphics, Symbolics

ATTRIBUTES

MACSYMA combines symbolic, numerical and graphical mathematics in one software package, with over 1,500 documented commands, and 600 executable demonstrations. MACSYMA's features include: Algebra - arithmetic, basic algebra, trigonometry, matrix algebra, solving equations (exact, approximate, numerical), sums and products, special functions. Calculus - differentiation and limits, Taylor series methods, indefinite integration, integral transforms, first and second order O.D.E.'s, systems of linear O.D.E.'s, perturbation methods, and numerical solutions. Vector and Tensor Analysis - vector calculus (dot and cross products, grad, div, curl, Laplacian operators, and many simplification options. Graphics - 2-D and

3-D plots, and other plot utilities. Utilities - pattern matching, properties data base, foreign language interface (FORTRAN and C code manufacturing), TeX output generator, and can also generate complete programs from symbolic specs.

ADDITIONAL REQUIREMENTS

SOURCE

MACSYMA Inc. (formerly SYMBOLICS Inc.)
20 Academy Street
Arlington, MA 02174
(617)646-4550

MAPLE

Symbolic Manipulation

APPLICATION

MAPLE is a powerful interactive system for algebraic manipulation or symbolic computations. MAPLE's facilities include arithmetic with integers, fractions, unknown variables, polynomials and general expressions, solving equations, factoring, taking derivatives and series expansions of functions, indefinite and definite integration, solving differential equations, matrix operations, plotting, and more.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Amiga DOS, Macintosh,
Sun 4/SPARC, IBM VM/CMS

ATTRIBUTES

MAPLE has an extensive library of approximately 2,000 procedures to provide access to the mathematical operations commonly used in science and engineering. Specific applications include VLSI design, relativistic physics, satellite guidance systems and electrical engineering. Some of the packages currently implemented in the MAPLE libraries include linear algebra, number theory, statistics, group theory, linear optimization, differential forms, and student calculus. An important property of MAPLE is that most of the algebraic facilities in the system are implemented using the high-level user language. The basic system, or kernel, is sufficiently compact and efficient to be practical for use in a shared environment or on personal computers with as little as 2 megabytes of main memory. The MAPLE kernel can produce input for C, FORTRAN, TeX, and a number of other standard systems.

ADDITIONAL REQUIREMENTS

SOURCE

Waterloo Maple Software
160 Columbia Street W.
Waterloo, ON N2L 3L3 Canada
(519)747-2373

MATHEMATICA

Symbolic Manipulation

APPLICATION

MATHEMATICA, a general system for doing mathematics by computer, is used as an interactive calculation tool and as a programming language. MATHEMATICA does numerical, graphical, and symbolic calculations. Capabilities include arbitrary precision arithmetic, special function evaluation, matrix manipulation, symbolic computation, and formulas manipulation directly in algebraic form. MATHEMATICA contains a core of mathematical knowledge, which can be extended by creating programs in the MATHEMATICA language. The language incorporates several hundred programs for numerical, symbolic, and graphical programming, and has extensive graphic capabilities.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Macintosh, Sun 4/SPARC, IBM RISC/6000, CONVEX C, SGI, and others

ATTRIBUTES

MATHEMATICA has over 600 built-in functions to support numerical and symbolic computation, as well as 2-D and 3-D graphics. The system permits users to create their own libraries for specialized computing environments. MATHEMATICA can evaluate a range of mathematical functions, including calculations using exact integers, rationals, complex numbers, and all standard special functions of mathematical physics. Symbolic computational abilities include polynomial operations, rational functions operations, calculus, equation solving, symbolic matrix operation, list operations, and tensor operations. The kernel works the same on all computers, and can produce input for C, FORTRAN, TeX, and other standard systems.

ADDITIONAL REQUIREMENTS

SOURCE

Wolfram Research Inc.
P.O. Box 6059
Champaign, IL 61826
(217)398-0700, (800)441-MATH

LOGICAL DECISION

Systems Engineering

APPLICATION

LOGICAL DECISION performs quantitative decision analysis. Based on a set of alternatives and a set of quantitatively measured characteristics to evaluate the alternatives, LOGICAL DECISION systematically formulates the preferences about the characteristics and then combines the information to provide a quantitative ranking of the selected alternatives. LOGICAL DECISION is capable of handling complex preferences, and alternatives with uncertainties, defined and described the way that makes the best sense to the user. Powerful methods from the field of decision analysis are used to quantify the preferences. Interactive features of the software permit evaluation of the alternatives instantly once alternatives and user preferences have been quantified.

SYSTEM AVAILABILITY

IBM PC/XT/AT or 100 percent compatible

ATTRIBUTES

LOGICAL DECISION provides the following features:

*Provides freedom in defining evaluation measures. Scales can be discrete or continuous, increasing or decreasing, with no restrictions on the range or number of scale points allowed.

* Handles up to 500 alternatives and 100 evaluation measures.
* Allows the description of alternatives using probabilities.
* Uses graphical assessment of preferences concerning single and multiple measures.
* Allows consistency checking and of preferences.

ADDITIONAL REQUIREMENTS

SOURCE

Logical Decisions
164 E. Scenic Avenue
Point Richmond, CA 94801
(415)233-8920

MICROSOFT PROJECT

Systems Engineering

APPLICATION

MICROSOFT PROJECT assists in project organization and management, combining the power of CPM (Critical Path Method) scheduling with a direct access to graphical environment. Direct Graphics allows flexibility and control needed to create and organize a project. With the reporting capabilities of MICROSOFT PROJECT, one can create, modify, and print quality reports. MICROSOFT PROJECT supports the transfer of information to other applications.

SYSTEM AVAILABILITY

286 (or higher) based MS-DOS PC's, Macintosh

ATTRIBUTES

The three main features used in MICROSOFT PROJECT to create, manage, and report a project are: Views, Tables, and Filters. Views is used to enter, organize, and examine tasks or resources. Using the Views menu, one can switch to the view that is best for the application. The Views menu consists of "traditional" project management charting tools: Gantt Chart, PERT Chart, Resource Form, Resource Graph, Resource Sheet, Resource Usage View, Task Entry View, Task Form, Task PERT Chart, and Task Sheet. By choosing different views from the Views menu, one may determine the desirable display of the project information. By applying Table or Filter to a view, the exact appearance of the information may be determined. By applying different tables, one can change the columns of information that appear in the views, while using a filter one can either display or highlight only the tasks or resources of interest.

ADDITIONAL REQUIREMENTS

SOURCE

Microsoft Corporation
One Microsoft Way
Redmond, WA 98052
(800)426-9400

RDD-100 SYSTEM DESIGNER

Systems Engineering

APPLICATION

Requirements Driven Development (RDD) is a software tool suite which encompasses all aspects of system design. In RDD, system behavior can be observed at all stages of the design process. RDD supports requirements analysis, functional design and specifications, allocation to subsystems and computers, and design of interfaces. RDD's graphic language describes real, dynamic system behavior, and includes constructs for integrated description of functions, conditions, interfaces, and flow of data or materials with control. RDD has a simulator which directly executes the design objects.

SYSTEM AVAILABILITY

386 and 486 based MS-DOS PC's, Macintosh, Sun 4/SPARC, HP-UX807, DEC workstation

ATTRIBUTES

RDD-100 System Designer facilitates the construction, maintenance, display, and documentation of design objects that specify behavior. Objects include requirements, functions, components, and the sources and decisions that lead to the system architecture. Objects are created and edited by graphics or text, with multiple generated views available to gain different perspectives. At the core of RDD is an extensible object base of data for system description, archiving and management, which follows the element-relationship attribute schema. Templates and consistency checks verify system design sufficiency.

ADDITIONAL REQUIREMENTS

SOURCE

Ascent Logic Corporation
180 Rose Orchard Way, Suite 200
San Jose, CA 95134
(408)943-0630

TEAMWORK

Systems Engineering

APPLICATION

TEAMWORK provides an integrated Computer-Aided Software Engineering (CASE) environment for system and software automated development. TEAMWORK provides a multi-user development system that takes advantage of advanced engineering workstation technologies such as interactive computer graphics, high performance servers, and industry standard, heterogeneous networks. The TEAMWORK environment provides the foundation for complete, concept to code CASE solution based on the unified CASE strategy. The TEAMWORK project database provides the foundation for every development project, particularly for project teams that use network workstations. Project information is stored in a central project database. The project data base contains the specifications and design models, the project data dictionary, project management data, project notes (both textural and graphic), documentation, and version and configuration details.

SYSTEM AVAILABILITY

IBM/OS-2, Sun 4/SPARC, HP300, HP400, HP700, VAX/VMS, Silicon Graphics, and others

ATTRIBUTES

The entire TEAMWORK product family shares an extensible, open architecture for seamless integration with other CASE tools, and is identical on all hardware platforms. Some of these toolkits are: IM (information modeling) for the analysis phase; SA (structured analysis); RT (real time) to enable analysts and designers to create, store, review, and maintain structured system specification; SIM (simulation and modeling); and others.

ADDITIONAL REQUIREMENTS

SOURCE

CADRE Technologies Inc.
222 Richmond Street
Providence, RI 02903
(401)351-CASE, (800)743-2273

TIMELINE 4.0

Systems Engineering

APPLICATION

TIMELINE is a PC class program/project management software that supports PERT, resource leveling, and detailed scheduling.

SYSTEM AVAILABILITY

IBM XT/AT or 100 percent compatible

ATTRIBUTES

TIMELINE performs WBS/OBS, link schedules, critical path, block time, lag time, dependencies, and resource leveling As Soon As Possible & As Late As Possible tasks. The software can handle 1,000 tasks per schedule, with task determined over minutes, hours, or days. TIMELINE can also support lead/lag dependencies, partial resource allocations, and sub-schedule roll ups.

ADDITIONAL REQUIREMENTS

SOURCE

Symantec Corporation
10201 Torre Avenue
Cupertino, CA 95014
(408)253-9600

Appendix D

SAMPLE DECISION ANALYSIS

SUMMARY

This Appendix provides an example of structured multi-attribute decision analysis.

A simple example of multi-attribute decision analysis follows. Assume that a project is operating with three technical performance measures, plus cost and schedule. The equation for this case may be written:

SCORE = WTM1*TM1 + WTM2*TM2 + WTM3*TM3 + WT*T + WC*C

where:

WTM1 = weight factor for technical performance measure #1
TM1 = normalized value of technical performance measure #1
WTM2 = weight factor for technical performance measure #2
TM2 = normalized value of technical performance measure #2
WTM3 = weight factor for technical performance measure #3
TM3 = normalized value of technical performance measure #3
WT = weight factor for schedule (time)
T = normalized value of schedule (development time)
WC = weight factor for cost
C = normalized value of cost

Assume that Option 1 in this study has the following values on each performance measure and that the range of estimates for all other options are as shown:

	Option 1 Projections	Range of Values among all Options
TM1 -	2400 bps (Data rate)	(1500 bps to 3000 bps)
TM2 -	97% (Accuracy)	(95% to 98%)
TM3 -	50m (Resolution)	(5m

to 130m)

T - 36 months (Development time) (30 months to 50 months)

C - $5,000,000 (Life cycle cost) ($3,500,000 to $7,000,000)

Normalization of values occurs by assigning the score of 1 to the best projected value on each performance measure and 0 to the worst expectation. All other predictions are then interpolated to their intermediate values between 0 and 1. Thus the normalized predictions for Option 1 would be:

Option 1 Projections Normalized Score

TM1 - 2400 bps (Data rate)
(2400-1500)/(3000-1500) = 0.60

TM2 - 97%
(0.97-0.95)/(0.98-0.95) = 0.66

TM3 - 50m
(130-50)/(130-5) = 0.64

T - 36 months
(50-36)/(50-30) = 0.70

C - $5,000,000
(7-5)/(7-3.5) = 0.57

Note that the value subtracted from the Option 1 estimate in the numerator of each equation is the estimate of the worst option on that performance measure. Thus, in the cases of technical performance measures one and two, the lowest estimate is used since more is better. Conversely for TM3, schedule and cost, less is better and the highest estimate among all options is subtracted in the numerator.

In order to finalize the score for Option 1, the relative weights of the performance measures are needed. Assume for this example that these values were given by the customer as:

Performance Measure Relative Importance

TM1 - Data rate
0.20

TM2 - Accuracy
0.35

M3	- 0.20	Resolution
T	- 0.10	Development time
C	- 0.15	Life cycle cost

This list indicates that accuracy is of the greatest concern, and development time for this range of values is relatively unimportant. Now the score for Option 1 may be calculated as:

SCORE(1) = 0.20*0.60 + 0.35*0.66 + 0.20*0.64 + 0.10*0.70 + 0.15*0.57

SCORE(1) = 0.63

The scores for all other options may be calculated in the same manner. A complete treatment of multi-attribute decision analysis has been derived by Keeney and Raiffa, "Decisions with Multiple Objectives: Preferences and Value Tradeoffs," which is listed in Appendix E, "References," "Selected References."

REFERENCES

SUMMARY

This Appendix provides a selected and annotated list of references and directives which are pertinent to systems engineering and project management.

NASA Policy Directives

NPD 2820.1, NASA Software Policies

NPD 7000.3D, Allocation and Control of Agency Resources

NPD 7120.4A, Program/Project Management

NASA Procedures and Guidelines

NPG 7120.5A, NASA Program and Project Management Processes and Requirements

NASA Internal Publications

SED Engineering Handbook EHB-1, Systems Engineering Division Product Assurance Plan, Langley Research Center, Systems Engineering Division, January 1990

NASA Systems Engineering Handbook, SP-6105, June 1995.

TL-790.M57 1992 The NASA Mission Design Process - A Guide to the Concept, Mission Analysis, and Definition Phases, Draft, Goddard Space Flight Center, Engineering Directorate, December 1992

MSFC-HDBK-1912, Systems Engineering Handbook - Volume 1 - Overview and Processes, George C. Marshall Space Flight Center, Systems Analysis Division, December 1994

MSFC-HDBK-1912, Systems Engineering Handbook - Volume 2 - Tools, Techniques, and Lessons Learned, George C. Marshall Space Flight Center, Systems Analysis Division, December 1994

Langley Policy Directives

LAPD 7120.2, Authority and Responsibilities of Managers of Small Space Flight Projects

Langley Procedures and Guidelines

LAPG 5000.2, Procurement Initiator's Guide

LAPG 5300.1, Space Product Assurance

LAPG 7320.1, Engineering Drawing System

Selected References

1. Barthelemy, Jean-Francois M., ed.: Recent Advances in Multidisciplinary Analysis and Optimization, Parts 1-3. NASA Conference Publication 3031, 1989. Proceedings of a symposium on multidisciplinary analysis held at Langley Research Center in September 1988.

2. Blanchard, Benjamin S., and Fabrycky, Wolter J.: Systems Engineering and Analysis. Second Edition, Prentice-Hall, Inc., 1990.

3. Charette, Robert N.: Software Engineering Risk Analysis and Management. Intertext Publications/McGraw-Hill Book Company, 1989.

4. Gibson, J. E.: How to do Systems Analysis, 1989.

5. Griffin, Michael D., and French, James R.: Space Vehicle Design. American Institute of Aeronautics and Astronautics, 1991.

6. Hoban, Francis T., and Lawbaugh, William M., eds.: Readings in Systems Engineering. NASA SP-6102, 1993.

7. Keeney, Ralph L., and Raiffa, Howard: Decisions with Multiple Objectives: Preferences and Value Tradeoffs. John Wiley and Sons, 1976.

8. Kerzner, Harold: Project Management - A Systems Approach to Planning, Scheduling, and Controlling. Second Edition. Van Nostrand Reinhold Company, 1984.

9. Lacy, J. A.: System Engineering Management: Achieving Total Quality. McGraw Hill, 1992.

10. Lewis, E. E.: Introduction to Reliability Engineering. John Wiley and Sons, 1987.

11. Phadke, Madhav S.: Quality Engineering Using Robust Design. PTR Prentice Hall, 1989.

12. Pittman, Bruce: Dynamic System Engineering: A Guide to Understanding and Implementing System Engineering. Pittman and Associates, 1991.

13. Rumbaugh, James; Blaha, Michael; Premerlani, William; Eddy, Frederick; and Lorensen, William: Object-Oriented Modeling and Design. Prentice Hall Inc., 1991.

14. Sage, Andrew P.: Systems Engineering. John Wiley and Sons, 1992. Focuses on systems methodology, design, and management and emphasizes the process of systems engineering rather than the product.

15. Walters, J. M., et al.: Addendum to the LASE Follow-on Pre-Phase A Study, 1993.

16. Wymore, A. Wayne: Model-Based Systems Engineering. CRC Press, 1993.

17. Anonymous: System Engineering Process. Center for Systems Management, Inc., 1993.

INDEX

INDEX - Continued

INDEX - Continued